Technical Report

IRI-TR-03-09

Development of a Simulation Environment for Water Drinking Networks: Application to the Validation of a Centralized MPC Controller for the Barcelona Case Study

Elena Caini*
Vicenç Puig**
Gabriela Cembrano

Institut de Robòtica i Informàtica Industrial

Abstract

In this report, MPC strategies have been designed and tested for the global centralized control of a drinking water network. Test have been implemented by using a software tool called PLIO, which allows the user to select the simulation parameters as well as the demands episodes in order to obtain the desired results. Additionally, this report describes the implementation of a MATLAB-based simulator of a plant model related to the Barcelona drinking water network. Several simulations and test have been done and conclusions from the obtained results are outlined and discussed.

* Elena Caini is with the Department of Information Engineering, University of Siena, Via Roma 56, 53100 Siena, Italy.

** Vicenç Puig is with the Control Department (ESAII), Technical University of Catalonia (UPC), Rambla de Sant Nebridi, 10, 08222 Terrassa, Spain, e-mail: vicenc.puig@upc.edu

Institut de Robòtica i Informàtica Industrial (IRI)
Consejo Superior de Investigaciones Científicas (CSIC)
Universitat Politècnica de Catalunya (UPC)
Llorens i Artigas 4-6, 08028, Barcelona
Spain

Tel (fax): +34 93 401 5750 (5751)
http://www-iri.upc.es

Corresponding author:

G. Cembrano
tel: +34 93 401 5785
gcembrano@iri.upc.edu
http://www-iri.upc.es/people/gcembrano

1 Introduction

Earth's surface consists of 70% water, but the 97.5% of water on Earth is salty water, while only the 2.5% is fresh water of which over two thirds is frozen in glaciers and polar ice caps. The remaining unfrozen freshwater is mainly found as groundwater, with only a small fraction present above ground or in the air. Fresh water is a renewable resource, although the world's supply of clean, fresh water is steadily decreasing. Water demand already exceeds the one supplied in many parts of the world, and as world population continues to rise at an unprecedented rate, many more areas are expected to experience this imbalance in the near future.
Sources where usable water may be obtained are:

- ground sources such as groundwater, aquifers, wells;
- precipitation which includes rain, hail, snow, fog, etc;
- surface water such as rivers, streams, glaciers;
- the sea through de-salination.

As a country's economy becomes richer, a larger percentage of its people tends to have access to drinking water and sanitation. Access to drinking water is measured by the number of people who have a reasonable means of getting an adequate amount of water that is safe for drinking, washing, and essential household activities. The drinking water is the water that is of sufficiently high quality so that it can be consumed or utilized without any risk of immediate or long term harm. Such water is commonly called *potable water*. In most developed countries, the water supplied to households, commerce and industry is all of drinking water standard even though only a very small proportion is actually consumed or used in food preparation (often 5% or even less). As it is easy to understand, the drinking water management in urban areas is a subject of increasing concern as conurbations growth.

Water supply networks are part of the master planning of communities, counties, and municipalities. Their planning and design requires the expertise of city planners and civil engineers, who must consider many factors, such as location, current demand, future growth, leakage, pressure, pipe size, pressure loss, etc. The advent of these systems, along with comparable sewage systems, was one of the great engineering advances that made urbanization possible. Improvement in the quality of the water has been one of the great advances in public health.
Like electric power lines, roads, and microwave radio networks, water systems may have a loop or branch network topology, or a combination of both. The piping networks are circular or rectangular. If any one section of water distribution mains fails or needs repair, that section can be isolated without disrupting all users on the network. While each zone may operate as a stand-alone system, there is usually some arrangement to interconnect zones in order to manage equipment or system failures.
In many cities where the conurbation has been growing fast and stormy rains are frequent, the existing combined sewer systems are unable to carry all the rain and the wastewater to the treatment plants when hight-intensity rain occurs. This result in flooding of certain areas and combined sewer overflows which release untreated water to the environment. Is simple to understand that this issue has an important impact in environmental and social areas.
Limited water supplies, conservation and sustainability policies, as well as the infrastructure complexity for meeting consumer demands with appropriate flow pressure and quality levels make water management a challenging control problem. Decision support systems provide useful guidance for operators in complex networks, where resources management best actions are not intuitive. Optimization and optimal control techniques provide an important contribution

to strategy computation in drinking water management, as reported in [16]; [12]; and [15].
Water systems are usually comprised of :

- **Supplies**, where raw water is drawn from superficial or underground sources, such as rivers, reservoirs or boreholes;

- **Production facilities**, where water is treated to meet consumer-use standards;

- **Transport systems**, consisting of canals and other natural or artificial open flow conduits which carry water from the sources to the treatment sites and to the distribution areas;

- **Distribution areas**, including consumer demands, storage tanks and pressurized pipe networks, to which water must be supplied with appropriate pressure levels;

- **Pressure and flow control elements** in all the above-mentioned subsystems, which make possible to meet demands with the available resources.

These systems are composed of a large number of interconnected pipes, reservoirs, pumps, valves and other hydraulic elements which carry water to demand nodes from the supply areas, with specific pressure levels to provide a good service to consumers. The hydraulic elements in a network may be classified into two categories: **active** and **passive** one. The active elements are those which can be used to control the flow and the pressure of water in specific parts of the network, such as pumps, valves and turbines. The pipes and reservoirs are passive elements, as they receive the effects of the operation of the active elements, in terms of pressure and flow, but they cannot be directly acted upon.
The topology of the network determines how an action in a certain element of a water network affects the rest. For example, in some simple tree-like networks an action at the end of a final branch may not affect the rest of the network at all and the sense of the flow into the elements is fixed, while in a mesh-like network, a more global influence of the actuation of most of the hydraulic control elements is expected. In fact a mesh-structure network contains several sources and it is highly interconnected so that the demands can be supplied from more than one source and, in general, the sense is not fixed in some of the valves or pipes. The topology of the networks is usually an important factor to be taken into account for the selection of more or less de-centralized schemes for the supervisory control system in general and for the control strategy optimization in particular.
Optimal control in water networks deals with the problem of generating control strategies ahead of time, guaranteeing a proper service of the network, while achieving certain performance goals, which may include minimization of supply and pumping costs, maximization of water quality, pressure regulation for leak prevention, etc.

2 Drinking Water Network Mathematical Model

First of all, in the analysis of the water networks, and in particular in their control, it is necessary to provide a complete description of the model elements. In fact, in model-based control techniques, like that of *predictive control*, the achievement of acceptable performance and satisfactory results mostly depend on the accuracy of the open-loop model. However, it is important to consider the trade-off between model accuracy and model complexity during its implementation and analysis.

2.1 Network Description

The water network structure establishes flow and pressure relationships between different elements, like, for example, mass conservation at a junction (node) or energy conservation in a closed loop. Additionally, a water network system contains a lot of flow (or pressure) control elements, controlled by the telecontrol system. These represent the active elements into the network, the so-called actuators. A systematic description of the dynamical model of the water network is achieved by considering the set of the flows through these m control elements, as the vector of control variables: $u \in \mathbb{R}^m$.
The state of the model is otherwise observed in the passive elements, such as the water storage tanks. Then, the set of the n reservoirs represents the vector of model state variables: $x \in \mathbb{R}^n$.
The demand sectors are considered like a stochastic disturbance in the model. Then, $d \in \mathbb{R}^q$ is a vector of know disturbances containing the values of the q demand sectors in the network. In order to use this model for predictive control, d should generally be a vector of demand forecasts, obtained through appropriate demand prediction models, based on the real data.
The dynamic model of the network could be written, in discrete time, as:

$$x(k+1) = f(x(k), u(k), d(k)) \tag{1}$$

This expression describes the effect on the network, at time $k+1$, produced by the control action $u(k)$ and the prediction demand $d(k)$ when the network state is $x(k)$. The function f represents the mass and energy balance in the water network and k denotes the instantaneous values at sampling time.
In many drinking water systems, the sampling time used for the control is one hour.
In the supervisory control system, the optimal control procedure receives informations about the current state of the network through the SCADA (Supervisory Control and Data Acquisition) system. The main information which the SCADA provides are:

- storage volume of water in every tanks;
- status of the pumps and valves;
- latest demands readings;
- pressure and/or flow values readings at selected points.

The optimization module contains a hydraulic model of the network which allows to test the effects produced by a control action on the network in terms of:

- water volume in the tanks;
- pressure and/or flow readings at selected points.

The optimal control procedure selects optimal strategies for the controllers of the active hydraulic elements, by searching in the space of possible controls and evaluating different alternatives. In water networks, where storage in tanks must be planned ahead to meet the future demands with specific pressure levels, the optimization involves the generation of controller strategies over a time period, called the optimization horizon, which may consist of one day, at hourly intervals, in a case of water distribution utility.
Considering all these things, it is now useful to give a detailed description of the different dynamic model of each hydraulic element. In addition, for every network element is set an operative range, for example the bounds for flow and pressure in the pipes or the volume in reservoirs are defined.

2.1.1 Tanks

The tanks represent the state of the model. Their dynamics are governed by the mass balance established between the volume and the flow in input and in output. The difference equation which describes the tank dynamical evolution is:

$$V_i(k+1) = V_i(k) + \Delta t(\sum_{h=1}^{n} q_{i,h}(k) - \sum_{j=1}^{m} q_{i,j}(k)) \tag{2}$$

where:

- V_i is volume of i^{th} tank;
- $q_{i,h}(k)$ is the h^{th} input flow in the i^{th} tank at instant k;
- $q_{i,j}(k)$ is the j^{th} output flow of the i^{th} tank at instant k;
- Δt is the discretization step that corresponds to the control sampling time.

Taking into account the geometry of the tank, a relationship between the volume and the level can be established.

2.1.2 Pressurized pipe

The flow in the pipes is related to head-loss between the extremes. This relationship is usually modelled through well-known non-linear approximations, such as the Hazen-Williams, Darcy-Weissbach, Colebrook-White equations (see [8], [9]):

$$q_{i,j}(k) = c_{i,j}(h_i(k) - h_j(k))^l \tag{3}$$

where:

- $q_{i,j}$ is the flow through a pipe between nodes i and j;
- $h_i(k)$ and $h_j(k)$ are the head values at nodes i and j respectively, at time k;
- $c_{i,j}$ is a parameter depending on pipe characteristics;
- l is the exponent representing the non-linearity of this relationship.

2.1.3 Nodes

The dynamical of the nodes does not have a time dependent behaviour. In fact, these elements could be considered as simple constraints. It is possible to see their rule in the network like that of cross-road where the input traffic is equal to the output traffic.
Then, the only constraint, which has to be satisfied in the node, is the mass balance: the sum of the input flows must be the same of the output flows.
Considering the case where there are a set of n input pipes and a set of outputs m pipes in the node, the equation, that should be satisfied, at every instant time k is:

$$\sum_{i=0}^{n} q_{i_{in}}(k) = \sum_{i=0}^{m} q_{i_{out}}(k) \tag{4}$$

where

- $q_{i_{in}}(k)$ is the input flow in the pipe i at instant k;
- $q_{i_{out}}(k)$ is the output flow in the pipe i at instant k.

2.1.4 Actuators

The actuators, pumps and valves, are assumed to be locally controlled. The set-point values of the flow in these elements is selected by the optimal MPC controller. At the first glance, these two types of elements look the same, but there are some differences, regarding the economical cost.
This different cost is due to the fact that the valves function is only to regulate the flow in pipes which connect elements with the same ground elevation or from one with a bigger ground elevation to an other with a smaller one. The function of a pump is to push the water from one with a smaller elevation to another with a bigger elevation.

2.2 PLIO as a Water Modelling Tool

PLIO software is a tool which allows to simulate and optimize the drinking water networks. This tool has been developed by UPC[1] and AGBAR in a project previous to WIDE project and has been applied to the control of the water networks in Santiago de Chile and Murcia cities.
It has a graphical interface which allows to represent the water network, with every element and connection, and to set the predictive control goal. The tool allows the whole operative planning of water cycle including supply, production, transport and distribution network in real-time. PLIO has been developed using standard GUI (graphical user interface) techniques and object oriented programming using Visual Basic.NET[2] how it is explained in [5].
PLIO calls a commercial solver, GAMS[3], to determine the optimal solutions of the optimization problem associated to the predictive optimal control using non-linear programming techniques. In a real time operation, an optimization problem is solved with a sampling time of a hour. This tool allows to do a detailed network study and to elaborate a model that suitably represents the reality. Using this application, it is possible to draw the network model and all their elements could be parametrized.

2.2.1 PLIO Operative Model

The PLIO software has four operation modes: editor, simulation, monitoring and reproduction mode as it is shown in Figure 1 [1].

Editor Mode This mode allows to graphically build and parametrize the network using a palette of building blocks, to define the control objectives and to generate the optimization model equations. In PLIO there are many different elements in the libraries that allows to draw the network easily. These elements include tanks, water demand sectors, sensors and actuators. The user positions the elements in the model dropping and connecting them with pipes or aqueducts. Each element of PLIO has a number of proprieties grouped in trees which identify the element, parametrize its characteristics, provide goals to the optimizer and define links to

[1] Universitat Politcnica de Catalunya
[2] Microsoft, 2002
[3] GAMS, 2004

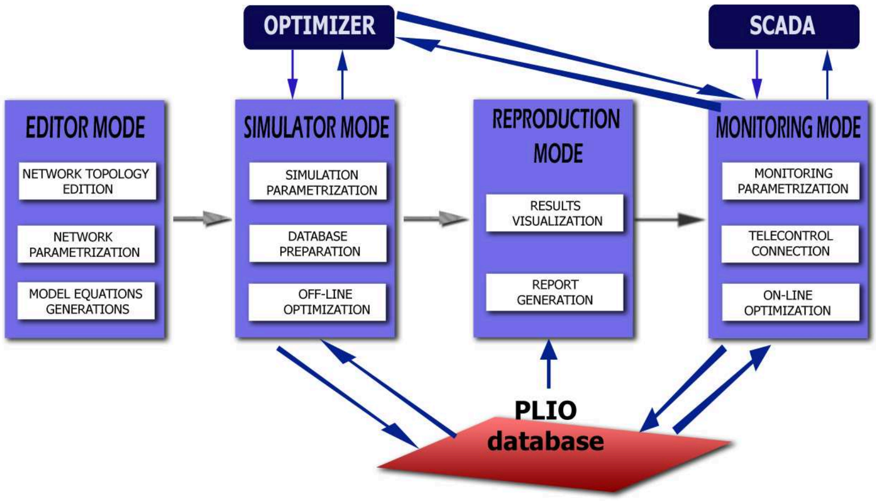


Figure 1: PLIO operating modes

SCADA and database. Once the network has been built, PLIO tests its consistency and connects it to the database. After the connection, it is necessary to synchronise PLIO model and the database. Moreover, PLIO generates the set of the optimization equations using goals and constrains defined in the proprieties of each element. In the window propriety, it is possible to set if an element is included or not in the optimization problem and its weight.

Simulation Mode The simulation consists in the off-line execution of a real scenario to check how the tool works in a water network. The simulations could be done with different forecast demand methods to analyze if the results improve or not. The demand used in this type of simulations are loaded in the database that is connected to the PLIO model. All the results, in terms of tanks volume, flow in the actuators or in the pipes, are then registered in the database so that it is also possible to draw their graphical evolution. At the end of the simulation, PLIO generates the optimal control using the GAMS solver.
It is also possible to set some parameters in the simulation windows propriety, like the starting and ending data, the interval between every iteration and the number of the iterations. This parameters are included in the particular scenario.

Monitoring Mode The optimization in real time (on-line) is executed in the monitoring mode. This is done using the demand and the measurements of the network real state, coming from the telemetry system, provided by SCADA system. PLIO generates the optimal controls, which are applied to the real network only after confirmation by an operator. Like in simulation model, graphical results of main network variables and controls can be represented and registered in the PLIO database for further studies.
In the monitoring mode there are four steps:

1. connection to SCADA;

2. data readings and writings;

3. optimization;

4. results treatment and graphical representation.

Steps number 2, 3 and 4 are repeated at each control cycle.

Reproduction Mode In this mode, it is possible to reproduce the state evolution under specific conditions and control set-points (optimal or other). This mode allows to see, in a simply way, through a graphical representation on the screen the values of flows or volume according to the selected element. Then, the represented value depends on the chosen element, and the display presents the element evolution for every reproduction iteration. Graphical interface could represent the main variable behaviours in a real or in a simulated scenario.

2.2.2 Dynamical Modelling of the Elements

The PLIO tool generates several variables and equations for every hydraulic element which allow to determine the model that describes the dynamical of the whole network. The type of variables and equations are different according to the elements role in the network. It is important to analyze the form of these equation in order to understand the optimization created by PLIO through the GAMS solver. For each element, in the following a detailed description of the equations and the variables is given. To simplify the understanding of this document the name of the variables are the same generated by PLIO.
In PLIO tool every variable has the unit of measurement according to the International System, thus, for example, the flows are given in m^3/s, while the volumes in m^3.

Tank The PLIO tool creates three positive variables for each tank: $\mathbf{V}xx$, $\mathbf{Vseg}xx$, $\mathbf{Vbajo}xx$, where xx corresponds to the name of the tank.
These variables are defined through three equations with the following names:

- $\mathbf{Vol}xx(t)$ corresponds to the instantaneous water level into the tank xx at time t. Its equation is:

$$\mathbf{Vol}xx(t+1)\ldots\mathbf{V}xx(t+1) = \mathbf{V}xx(t) + \sum_i Q_i^{in}(t) - \sum_i Q_i^{out}(t) \tag{5}$$

  where $Q_i^{in}(t)$ and $Q_i^{out}(t)$ are the input and output flows, respectively, of the tank xx at time t. The last variable present in the equation (5) is $\mathbf{V}xx(t)$ which represents the volume of the tank xx at time t.

- $\mathbf{Volseg}xx(t)$ corresponds to the security level in the tank and indicates if there is a penalty or not, according to:

$$\mathbf{Volseg}xx(t)\ldots\mathbf{Vseg}xx(t) = max(0, \mathbf{V}xx_{penalty} - \mathbf{V}xx(t)) \tag{6}$$

  where $\mathbf{V}xx_{penalty}$ is the volume value under which there is a penalty.
  The way to determine this penalty value will be discussed later in the report. This volume should be used to minimize the electrical and water costs to satisfy the demands. This equation allows to penalize the cost function only when the actual level is below the security one.

In fact, the *max* function in equation (6) imposes a penalty value $\mathbf{Volseg}xx(t)$ equal to zero when the volume at time t is over the security level.

- **Volbajo**xx corresponds to the normalization of the penalty value, that allows to compare the term coming from each tank, and is given by:

$$\mathbf{Volbajo}xx \ldots \mathbf{Vbajo}xx = \frac{\sum_{t=1}^{24}[\mathbf{Volseg}xx(t)]^2}{\left([\mathbf{V}xx_{penalty}]^2 * 24\right)} \tag{7}$$

where the 24 at the denominator of the function represents the normalization taking into account the prediction horizon. The summatory includes all prediction terms calculated for each instant of the prediction horizon, that it is set to 24 hours. Then, to obtain a normalized value it is necessary to divide by the same value. This term, logically, does not have any dependence on time, since it considers the whole time horizon of optimization.

Equation (7) corresponds to the security objective in the cost function.

Nodes The PLIO software tool creates for these variables only one equation, without generating any positive variables. This is reasonable because the node could be considered like a zero-volume tank where it is not possible to store water. The only thing that it is important in the node is that the input flow has to be equal to output flow. In this element type there is not the need to set any parameter.
The equation **Bal**xx, generated by PLIO, assures, precisely, that the nodes xx could not store water. In fact, it represents a mass balance of input and output flows for each instant time t:

$$\mathbf{Bal}xx \ldots \sum_i Q_i^{in}(t) = \sum_i Q_i^{out}(t)$$

where $Q_i^{in}(t)$ and $Q_i^{out}(t)$ are, respectively, the input and output flows of the node xx at the instant t.

Pumps The pumps in the network represent one type of the actuators and so they require a particular attention in their description. PLIO tool software creates, as for the tanks, three positive variable for each pumps: **Q**xx, **Est**xx and **sum**xx, where xx are, as usually, the name of the pump.
Using these variables 4 equations are generated which determine the behaviour of the pumps and, in particular, they describe: the flow both in input and in output, the economical cost and the stability.
Now their detailed description is presented:

- **Caud1**$xx(t)$ represents the pumps input flow:

$$\mathbf{Caud1}xx(t) \ldots \mathbf{Q}xx(t) = \mathbf{Q1}yy(t)$$

where $\mathbf{Q1}yy(t)$ is the flow into the pipeyy at time t, where the pipeyy is the pipe at the input of the pump xx.

- $\mathbf{Caud1}xx(t)$ represents the pumps output flow:

$$\mathbf{Caud1}xx(t)\ldots\mathbf{Q}xx(t) = \mathbf{Q2}zz(t)$$

where $\mathbf{Q2}zz(t)$ is the flow into the pipe zz at time t, where the pipe zz is the pipe at output of the pump xx.

- $\mathbf{Tot}xx$ is the cost equation of the pump xx:

$$\mathbf{Tot}xx\ldots\mathbf{sum}xx = \sum_{t=1}^{24}\Big(\mathbf{Q1}zz(t) * \mathbf{CE}xx(t) * \frac{1}{\mathbf{Q}_{max}xx}\Big) \tag{8}$$

where $\mathbf{Q}_{max}xx$ is the maximum admissible flow through the pipe xx (the maximal value for each pump is shown in the Table 2); $\mathbf{Q1}zz(t)$ is the flow through the pipe zz, which is the pipe at output of the pump xx, at time t; $\mathbf{CE}xx(t)$ is the electrical cost of the pump xx at time t. Dividing by $\mathbf{Q}_{max}xx$ a sort of normalization is done. In fact, in this way, it is possible compare the values coming from every pump. This term is the result of the cost in the whole prediction horizon. Notice that the equation (8) is not depending on time. This values is included in the price objective of the cost function.

- $\mathbf{Estab}xx$ is the stability component for pump xx. This equation, also as the cost component, is not time depending. In fact, both these equations have a summatory for the whole optimization time horizon:

$$\mathbf{Estab}xx\ldots$$

$$\ldots\mathbf{Est}xx = \frac{\big(\mathbf{Q}xx(0) - \mathbf{Qpast}xx\big)^2 + \sum_{t=1}^{24}\Big(\mathbf{Q}xx(t+1) - \mathbf{Q}xx(t)\Big)^2}{\big(\mathbf{Q}_{max}\big)^2 * 25} \tag{9}$$

where 25 is the normalization term of the sum of 25 elements. In fact, with the summatory, there are 24 additions, due to a 24 hours of prediction horizon, and also there is the component concerning the initial condition $\big(\mathbf{Q}xx(0) - \mathbf{Qpast}xx\big)$, which considers the gap with the resulting value of the previous iteration. The $\mathbf{Q}xx(0)$ is the flow through the pump xx at time 0, of this iteration. The equation (9) takes part in the cost function in the stability objective.

Valves The valves represent the other type of actuators present in the network. So, PLIO software deals with these elements in a similar way than the pumps. In fact, the variables and the equations created are very similar to those concerning the pumps. The only difference, between these two type of actuators, is about the cost function, considering that the valves have not an electrical cost coefficient.
Therefore each valve has, only, two positive variable, instead of the three of the pumps: $\mathbf{Q}xx$, $\mathbf{Est}xx$, the last one is created only in the stabilized valves, which are obtained using the motorized valves element.
The equations, which regulate the relationship between these two variables, are three:

- $\mathbf{Caud1}xx(t)$ represents the valve input flow at time t:

$$\mathbf{Caud1}xx(t)\ldots\mathbf{Q}xx(t) = \mathbf{Q1}yy(t)$$

where $\mathbf{Q1}yy(t)$ is the flow inside the pipeyy at time t, where the pipeyy is at the input of the valve xx.

- $\mathbf{Caud1}xx(t)$ represents the valve output flow at time t:

$$\mathbf{Caud1}xx(t)\ldots\mathbf{Q}xx(t) = \mathbf{Q2}zz(t)$$

where $\mathbf{Q2}zz(t)$ is the flow into the pipe zz at time t where the pipe zz is the pipe in output from the valve xx.

- $\mathbf{Estab}xx$ is the stability component for valve xx. This value, as in case of the pumps, does not depend on time, because there is the summatory which include the whole prediction horizon.

$$\mathbf{Estab}xx\ldots$$

$$\ldots\mathbf{Est}xx = \frac{\big(\mathbf{Q}xx(0) - \mathbf{Qpast}xx\big)^2 + \sum_{t=1}^{24}\Big(\mathbf{Q}xx(t+1) - \mathbf{Q}xx(t)\Big)^2}{\big(\mathbf{Q}_{max}\big)^2 * 25} \tag{10}$$

This equation and its normalization are equal to that explained in the equation (9) and both correspond to the stability term of the objective function.

Pipes Piper are modelled in PLIO by two positive variables that represent the flow at the input and output of the pipe xx: $\mathbf{Q1}xx$ and $\mathbf{Q2}xx$ The equation $\mathbf{Caud}xx(t)$ is very simple, it only imposes that both the flows $\mathbf{Q1}xx$ and $\mathbf{Q2}xx$ should be the same at every time t:

$$\mathbf{Caud}xx(t)\ldots\mathbf{Q1}xx(t) = \mathbf{Q2}xx(t)$$

In this model, it is not considered the constraints related to the pressure, because we have a valve or a pump with a flow controller in all pipes in the considered network configuration guaranteeing that the flow is established by the set-point of the MPC controller. Then, the pipes appear as a simple medium where the water runs.

Sources PLIO models the supply elements xx by one positive variable $\mathbf{sum}xx$ and two equations, which establish its behaviour:

- $\mathbf{Lim}xx(t)$ which represents the flow at time t in the source xx:

$$\mathbf{Lim}xx(t)\ldots\mathbf{Q}xx(t) = \mathbf{Q1}yy(t)$$

where $\mathbf{Q1}yy(t)$ is the flow of pipe yy in output from the source xx.

- $\mathbf{Tot}xx$ represents the total cost of the source:

$$\mathbf{Tot}xx\ldots\mathbf{sum}xx = 2\cdot unitary\_cost\cdot\sum_{t=1}^{24}\mathbf{Q1}yy(t) \tag{11}$$

where $\mathbf{Q1}yy(t)$ is the output flow of pipe yy at time t, and the *unitary_cost* is the coefficient cost inserted in the source property.

This coefficient indicates the cost of the water withdrawal from the source xx. The sources have different unitary_costs depending on the different elevations, treatments and paths to arrive to the users.

3 Model Predictive Control

Water supply and distribution systems are very complex multivariable systems. In order to improve their performance, *predictive optimal control* provides suitable techniques to compute optimal control strategies *ahead in time* for all the flow and pressure control elements.
The optimal strategies are computed by optimizing a mathematical function describing the operational goals in a given time horizon and using a representative model of the network dynamics, as well as demand forecasts.
The computation of optimal strategies must take into account the dynamics of the complete water system:

- 24-hour-ahead demand forecasts;
- 24-hour-ahead availability predictions in supply reservoirs and aquifers, defined by long-term planning for sustainable use;
- 24-hour-ahead predictions of production plant capacity and availability;
- current state of the water system provided by the telemetry system;
- 24-hour historic data in open channel sections, due to delays in water transport;
- physical and operational flow constraints in all the elements.

A model of a water system is a tool to predict the effect of control actions on all the network elements. For the purpose of on-line optimal control, a large number of control actions must be tested and evaluated during the optimization process. Therefore, it is important for the mathematical models developed to be:

· representative of the hydraulic dynamic response;

· simple enough to allow a large number of evaluations in a limited period of time, imposed by real-time operation.

An operational model of an urban drinking water network system is a set of equations which provide a fast approximate evaluation of the hydraulic variables of the network and its response to control actions at the gates. This type of model is useful for the computation of optimal strategies, because it makes possible to evaluate a large number of control actions in a short computation time.
One of the most used and effective control strategy for the drinking water control problem is the Model Predictive Control (MPC) ([3]; [2]; [11]; [10]).
The predictive controller usually deals with the middle level of a control structure where at the top we can find the modules that provide state estimation and the demands forecast over the control horizon. This information is the input into the MPC problem. The outputs of the MPC controller are reference values for the local controllers that implement the calculated set-points. The drinking water network has many control objectives and so also the optimization problem

associated with the MPC controller is a multiple objective as well.
Usually the approach to solve a multi-objective optimization problems is to form a scalar and linear cost function, composed by a weighted sum of cost functions associated with each objective. When the objectives have a priority, it is possible to select a bigger weight that represents the importance of the objective in the optimization. It is quite difficult to find the appropriate weights for every component in the cost function. In fact in every different scenario with different numerical value the appropriate weight could change. Moreover, the weights serve to normalize the cost functions as well as manage their priority.
An alternative to weight based method, is the lexicographic approach, see [13], which is based on assigning “a priori ”different priorities to the different objectives and then focus on optimizing the objectives in their priority order.

3.1 MPC Strategy

All the controllers in the MPC family are characterized by the receding horizon strategy shown in Figure 2, where N is the predictive horizon ([4]), that in the case of a drinking water network,

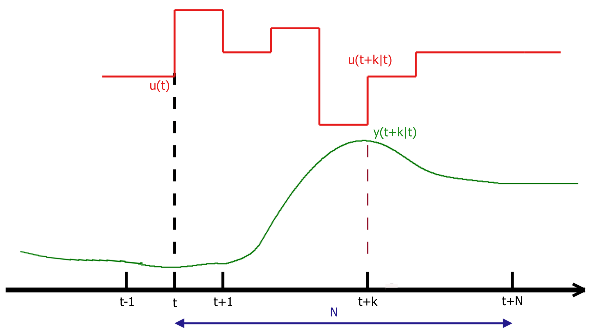


Figure 2: *MPC strategy*

usually, is set a 24 hours, as it is explained at the beginning of this chapter. In more detail:

1. A prediction of the N future outputs are calculated at each instant t:

$$\hat{y}(k+i|k)^4 \qquad \text{for} \quad i = 1, \cdots, N$$

   These outputs depend on the known values at instant t and on the future control signals:

$$\hat{u}(k+i|k) \qquad \text{for} \quad i = 1, \cdots, N-1$$

   which are the ones to be calculated.

2. The set of future control signals is calculated by optimizing a determinated criterion function in order to keep the process as close as possible to a reference trajectory (which can be the set-point itself or a close approximations of it). The most used criteria are based on a quadratic function error between the predicted output signals and the predicted reference trajectory.

[4]the notation indicates the value of the variable at the instant $k+i$ calculated at instant k

3. The control signal $u(k|k)$ is sent to the process while the next control signals calculated are rejected, because at the next sampling instant $y(k+1)$ is already known and the step 1 is repeated with this new value and all sequences are brought up to date. Thus, the control $u(k+1|k+1)$ is calculated (which in principle would be different to the control $u(k+1|k)$ because the new information is available) using the receding horizon concept. The basic structure of this strategy is shown in the Figure 3 where a model is used to predict the future plant outputs. The prediction is based on the past and current values and on the proposed optimal future control actions. These actions are calculated by the optimizer taking in account both the cost function and the constraints.

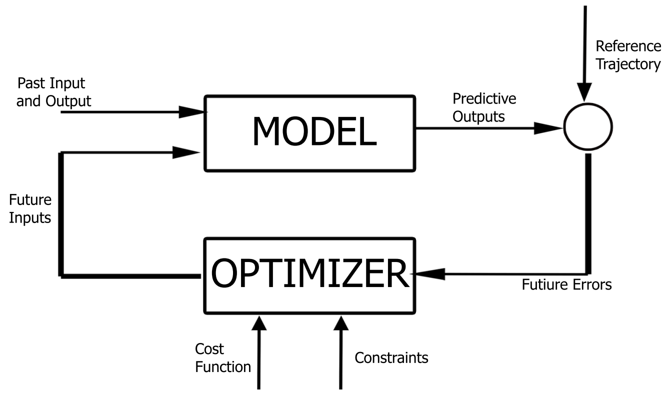


Figure 3: *Basic structure of MPC*

The process model plays an important role in the control. Indeed, the chosen model must be capable of capturing the process dynamics so as to precisely predict the future outputs as well as being simple to implement and to understand. Taking in account this, the formulation of the problem is a fundamental step in the building of MPC controller. MPC is not a unique technique but it could be seen like a set of different methodologies since there are many types of models used in various formulations.
Logically, the optimizer is another fundamental part of the strategy since it provides the control actions. If the cost function is quadratic, its minimum can be obtained as an explicit linear function. Otherwise, when there are some inequality constraints the solution it has to be obtained by more computationally demanding numerical algorithms.
The size of the optimization problems depends on the number of variables and the prediction horizon used. It is important to remind that the amount of time required in a constrained and robust case could be various order of magnitude higher than the one needed for the unconstrained case.

3.2 Problem Formulation

The formulation used in the majority of predictive control literature [10], is based on a linear system, on a quadratic cost function and on a linear inequalities constrains. Moreover, it is assumed that the model is time invariant.
The cost function used does not usually penalise particular values of the input vector $u(k)$, but only the changes of the input vector, $\Delta u(k)$. Then, it is considered the linear, discrete-time, invariant-time state-space model of the system:

$$\begin{cases} x(k+1) = Ax(k) + B_u u(k) + B_d d(k) \\ y(k) = Cx(k) \end{cases} \tag{12}$$

where: $A \in \mathbb{R}^{n \times n}$, $B_u \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{n \times n}$ are the state space matrices and $B_d \in \mathbb{R}^{n \times q}$ is a known disturbance, in this particular models the demands. Then x is a n-dimensional state vector collecting to the n tanks volume; u is an m-dimensional input vector which represent the flows in the actuators and $d(k)$ is a q different component corresponding to the q demand sectors. The index k counts the time step. So the sequence of actions at time step k is the follow, according to the strategy described before (Figure 2), are the following:

1. To obtain measurements $y(k)$;
2. To compute the required system input $u(k)$;
3. To apply $u(k)$ to the plant.

This implies that there is always some delay between measuring $y(k)$ and applying $u(k)$.
The constrains, presented in this model, are those concerning the physical operational limits of the elements:

$$\begin{cases} u_{i_{min}} < u_i(k) < u_{i_{max}} & \text{for} \quad k = 0, \cdots, N-1 \\ & \qquad\; i = 1, \cdots, m \\ y_{i_{min}} < y_i(k) < y_{i_{max}} & \text{for} \quad k = 1, \cdots, N \\ & \qquad\; i = 1, \cdots, n \end{cases} \tag{13}$$

3.2.1 Control Objectives

The various MPC algorithms propose different cost functions to obtain the control law. The general rule is that the future output on the considered horizon should follow a determinated reference signal and, at the same time, the control effort (Δu) necessary for doing this should be penalised.
A drinking water network has multiple objectives which could assume different priority [5], [6]. First of all, the main goal is that of satisfying the demands. Achieving this result, the predictive control strategy has also to take into account the optimization of the system performance in terms of different operational criteria. In general, the most common objectives are related to the physical limits of the elements to avoid their damage, or to the minimization of the economical cost.
In detail, the criteria which could considered are:

- **Security**: this criteria maintains the volume in the tank over a threshold in order to avoid infeasibilities.
- **Quality**: this objective is especially important when several sources exists with a different water quality, which could depend on the level or on the concentration of some ion that decays in time.
- **Stability**: this criteria aims to avoid continuous and abrupt set-point variations in the valves or pumps which means that all treatment plants and actuators operate as smoothly as possible. This point is very important to avoid damage in valves or pumps.

- **Price**: the electrical cost (price) in the network type is attributable to the water cost in the source and to the electrical cost necessary for the pumping. The water cost could be different at different sources with different elevation or treatment, while the electrical cost change depending to the hour of the day.
- **Conservation**: water sources such as reservoirs and rivers are usually subject to operational constraints to maintain water levels, ecological flows and a sustainable water use.

The control strategy selection at each time k consists of posing and solving an optimal control problem. This means finding the set of admissible controls (within the physical and operational constraints) which optimize performance index $J(x, u, d)$ over the optimization horizon. The performance index J is a general non-linear function of the state and control variables, which may contain:

- Non-linear, usually quadratic, penalty function for low storage tank levels, related to the **security** objective. Then, this term depends on the state $x(k)$. The component for every tank i (with $i = 1, \cdots, n$) at each instant k (with $k = 1, \cdots, N$) is called $Sec_i(k)$:

$$Sec_i(k) = \max\{0,\ x_i(k) - Vpen_i\}$$

  where $Vpen_i$ is the threshold volume selected for every tank i. To obtain a quadratic index the value $Sec_i(k)$ is squared, and after the index is normalized dividing by $[Vpen_i]^2$:

$$pen_i(k) = \frac{[Sec_i(k)]^2}{[Vpen_i]^2} \tag{14}$$

- Linear or non-linear time-varying cost for water acquisition and pumping. This term, related to some values of the input vector $u(k)$ is simply computed by multiplication between the flow and the hourly cost.
- Non-linear (quadratic) penalty function of abrupt changes in control actions. This term optimize the **stability** objective and it is related directly to the changes of the input vector $\Delta u(k)$. In fact, this term could be defined, for each instant k (with $k = 1, \cdots, N-1$) and for each input i (with $i = 1, \cdots, m$) as:

$$\Delta u_i(k) = [u_i(k) - u_i(k-1)] \tag{15}$$

  Like in the storage tank level in the cost function, it is better to consider the square value of Δu, with an appropriate normalization.
- Non-linear function of flow for quality regulation, related to the input vector.
- Other terms according to the operational goal.

The same normalization, as the one used in the equation (14), is necessary to allow to sum together these different objectives with different magnitudes.
More precisely, at each time step, the MPC strategy computes a control input sequence of present and future values:

$$[u(k), u(k+1)..., u(k+N-1)] \tag{16}$$

which allows to optimize an open-loop performance function, according to a prediction of the system dynamics over the horizon N. This prediction is performed using demand forecasts and the network model, described in the equation (1).
However, only the first control input of sequence $[u(k)]$ is actually applied to the system, until another sequence based on more recent data is computed.
The same procedure is restarted at time $k+1$, using the new measurements obtained from sensors and the new model parameters obtained from the recursive parameter estimation algorithm that is working in parallel.
The resulting controller belongs to the class called open-loop optimal-feedback control. As the name suggests, feedback from the telemetry system is used, and the optimal control strategy is re-computed at each time k.

3.2.2 Multi-Objective Optimization

Considering what it is said above, the optimization problem associated with the MPC controller is multi-objective.
The general ideas for this problem type could be formulated as the minimization with respect to every objective $f_i(k)$.
The functions $f_i(k)$ are obtained summing the costs introduced by every element which are included in i criteria at the instant k. For example, in the case of the security criteria, the function is obtained through the sum in equation (14):

$$f_1(k) = \frac{\sum_{i=1}^{n} pen_i(k)}{n} \tag{17}$$

where n is the number of the states in the system. In this way, the $f_1(k)$ is a normalized value. The normalization is done according to the square penalty for each tank in the term $pen_i(k)$, and to the number of the tanks to obtain the function $f_1(k)$.
A simple way to solve a multi-objective optimization is through scalarization. This means converting the problem into a single-objective optimization problem with a scalar-value objective function.
The most common form for a scalar objective function is a linearly weighted sum of the functions f_i, which represents every objective that has to be optimized, like for example the security objective in the equation (17):

$$F(k) = \sum_{i=1}^{r} \omega_i f_i(k) \tag{18}$$

where r is the number of objectives present in the problem.
The priority of the objectives are reflected by the weights ω_i: when there is a bigger weight the goal has a bigger priority.
For an evaluation over the entire value of the optimization horizon, the performance index has to be summed as:

$$J = \sum_{k=1}^{N-1} F(k) \tag{19}$$

where N is the optimization horizon, in a number of sampling periods.

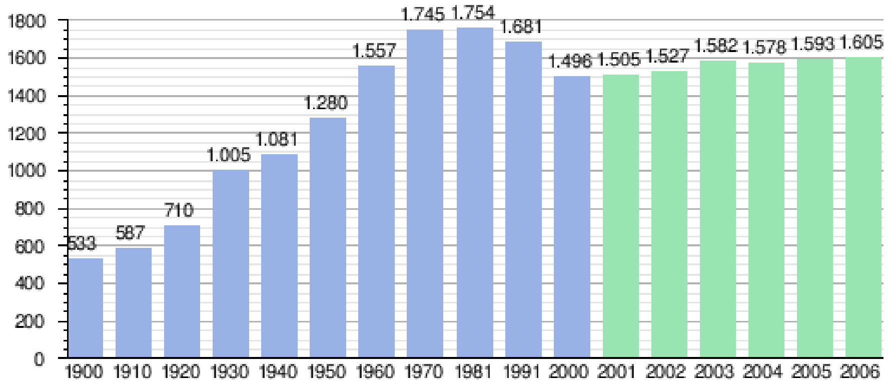


Figure 4: *Population behaviour of Barcelona city since 1900 to 2006*

4 Case Study: Water Barcelona Network Description

4.1 System Model

This report considers as a real application the case study of the Barcelona water network. First of all, it is important to notice the environment of the city that is presented to better understand the problem we are facing. Geographically, Barcelona has a strong slope in the zone near to the mountain, which decreases in direction towards the Mediterranean sea. The city of Barcelona has a population of about 1.605.000 in an area of 98 Km^2, that means a very high density of population (more than 16.000 per Km^2). The fast growth of the city during the XX century, as it is showed in Figure 4, has lead to improve the drinking water network continuously.

The weather of Barcelona is the typical of Mediterranean climate. The yearly rainfall is not very high (600 mm/year), but it includes heavy storms, rains with great intensity which could concentrate in thirty minute the fourth part of the yearly precipitation. The two last issues are interconnected. In fact the urban environment affect the local climate which cause a thermal difference between Barcelona and its surroundings. This difference could reach 3 or 4 Celsius degree. This phenomenon increases the intensity of the storm.

Considering all of these previous considerations, it is logical to think and understand that the drinking water network of Barcelona is a very complex interconnected system, as shown in Figure 5.

The water supply of Barcelona network is basically constituted of three sources: the Ter river through treatment station of Cardedeu, the superficial and the underground Llobregat river. The superficial Llobregat river came from the treatment stations of Abrera and Sant Joan Despí, while the underground water is stored in the aquifer of Llobregat delta. The source situations are shown in Figure 6.

The company which manages the distribution of the water is AGBAR[5]. This society initiated the automatisation of the water distribution network in 1969 and in 1976, a first centralized control system was installed . Since this date this system is being continually improved, like is possible to see in [14].

In 1984, AGBAR and UPC developed an MPC controller on-line but the size of the network implies real-time constraints that were not easily feasible to satisfy with the computer computation speed of that time. Later, in the 2002 AGBAR and UPC start the development of the PLIO tool that allows, as discussed in Chapter 2, the modelling and MPC control of water networks. Actually, there is a project running in parallel with WIDE project, named SOSTAQUA, whose

[5]Sociedad General De AGua de BARcelona, S.A.

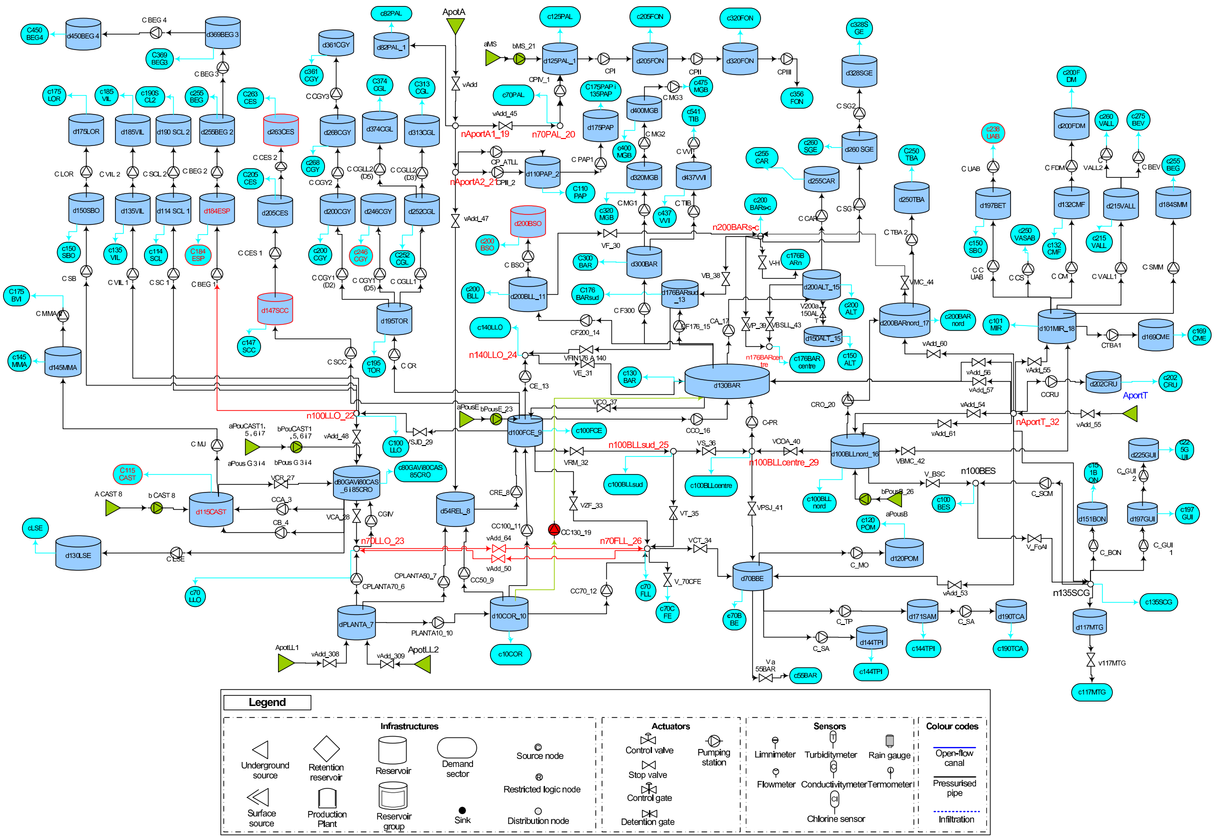


Figure 5: *Complete Barcelona network model*

aim is to apply PLIO to the Barcelona water network.
The water supply to the users is done through a complex distribution network which allows to provide the water at any different ground elevation, where usually there are several reservoirs to adapt pressure to ground topography and so it is possible to assure a supply with a correct pressure and quality.
The model studied in this report is a prototype of the Barcelona urban water network. The complete Barcelona water network is showed in Figure 5. The prototype network corresponds to an aggregated model of the real one. The Figure 7 shows the network conceptual model using PLIO modelling methodology. It is possible to observe the whole set of the hydraulic elements, like pumps, valves, tanks, pipes and sources.

The aggregated model has 9 sources, corresponding to:

- 4 superficial resources:
  - AportA which represents the water that come from the Abrera potabilisation station;
  - AportLL1 and AportLL2 which come from underground and superficial water from Llobregat river in the Sant Joan Despí potabilisation;
  - AportT which corresponds to the water coming from the Ter river treated by Cardedeu potabilisation station.
- 5 underground resources aMS, aPousB, aPousE, aCast, aPouCast.

In addition to this, the Figure 7 shows 17 tanks in sky blue. These elements are the state variables of the dynamical network model.

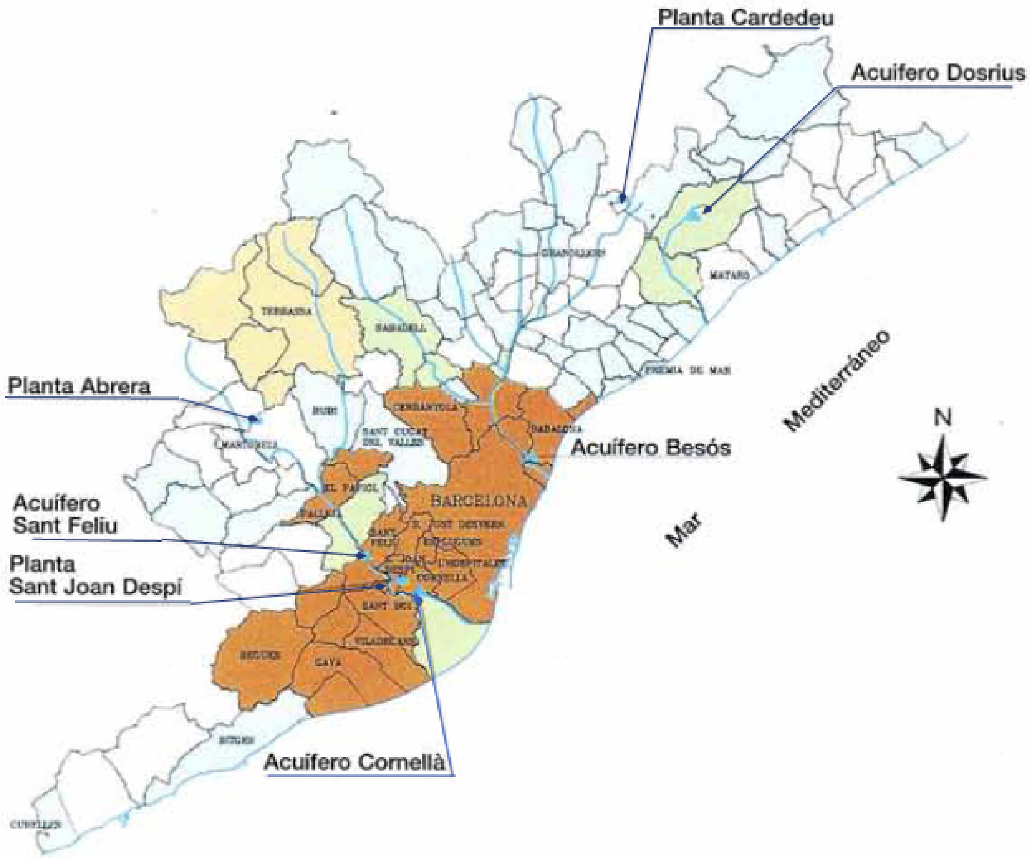


Figure 6: *Water supply map for the Barcelona network*

The actuators in the network are 61 and they are, in particular, composed of 26 pumps and 35 valves. Some of these actuators are used to control superficial and underground sources.
The 11 nodes, that appear in the network, are considered as constraints where the sum of input flow must be equal to the sum of output flow.
Finally, there are 25 demand sectors, in blue. The demand patterns have been provided by AGBAR society since they try to reflect the real demand as close as possible. In the model, demands are considered as known disturbances.

The network has been modelled through the user-friendly software tool (PLIO), that, how it is explained in the previous chapter, allows to simulate and optimize drinking water networks. The program generates an hydraulic optimization model, which is solved timely for real time implementation and also is useful as a decision support tool. The Figure 8 shows the PLIO graphical model of the aggregated Barcelona network.
Through PLIO tool, it is possible to look for a optimal solution, that means trying to minimize the economical costs while satisfying the whole demand. For this purpose, PLIO calls a commercial solver (GAMS), which determines the optimal solution of the optimization problem, associated to the predictive optimal control, using non-linear programming techniques. For the predictive control scheme, a prediction horizon of 24 hours is chosen, as usually it is done in the MPC control of drinking water networks. The solution found by GAMS solver should satisfy

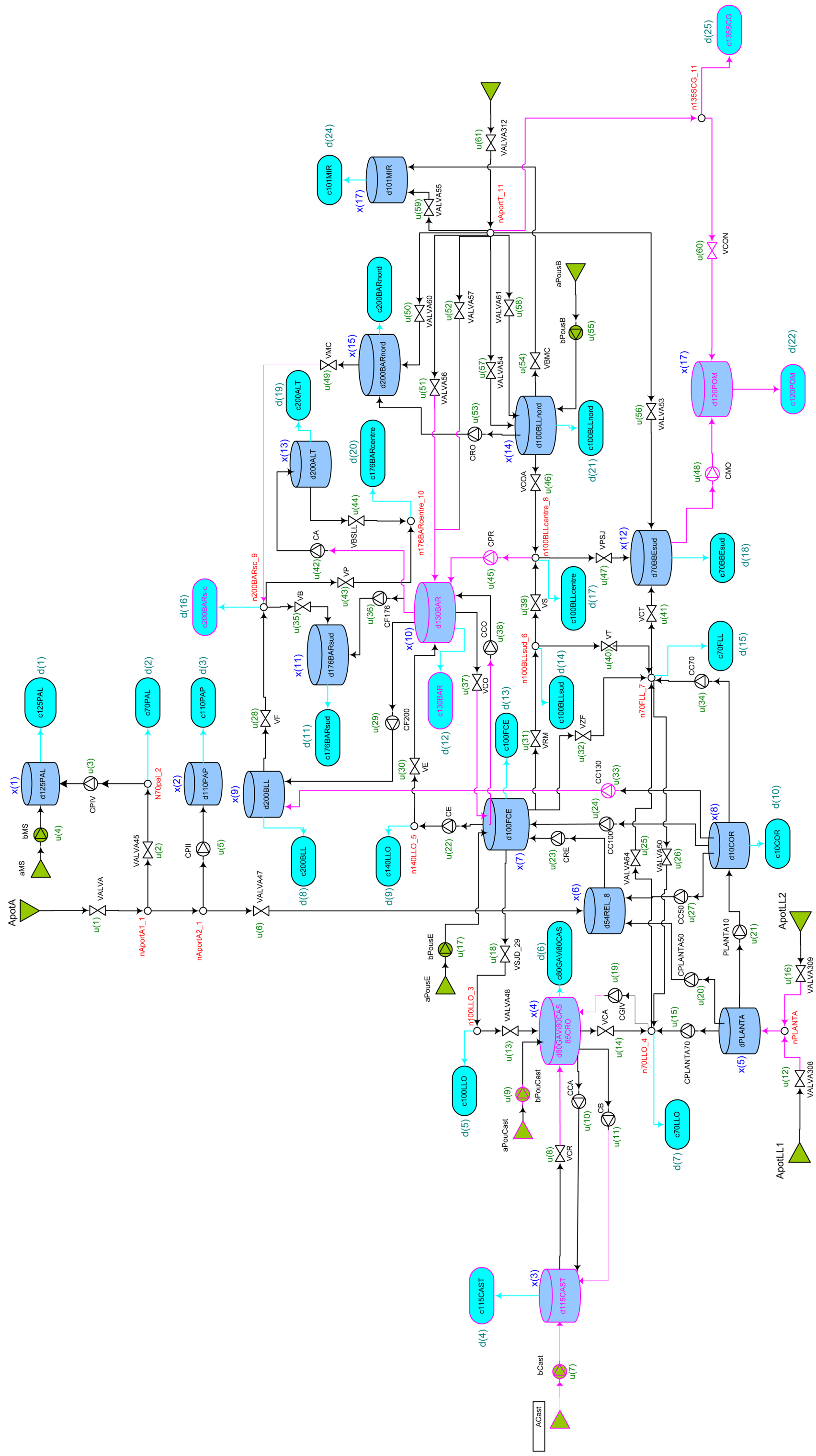


Figure 7: *Aggregated Barcelona model network*

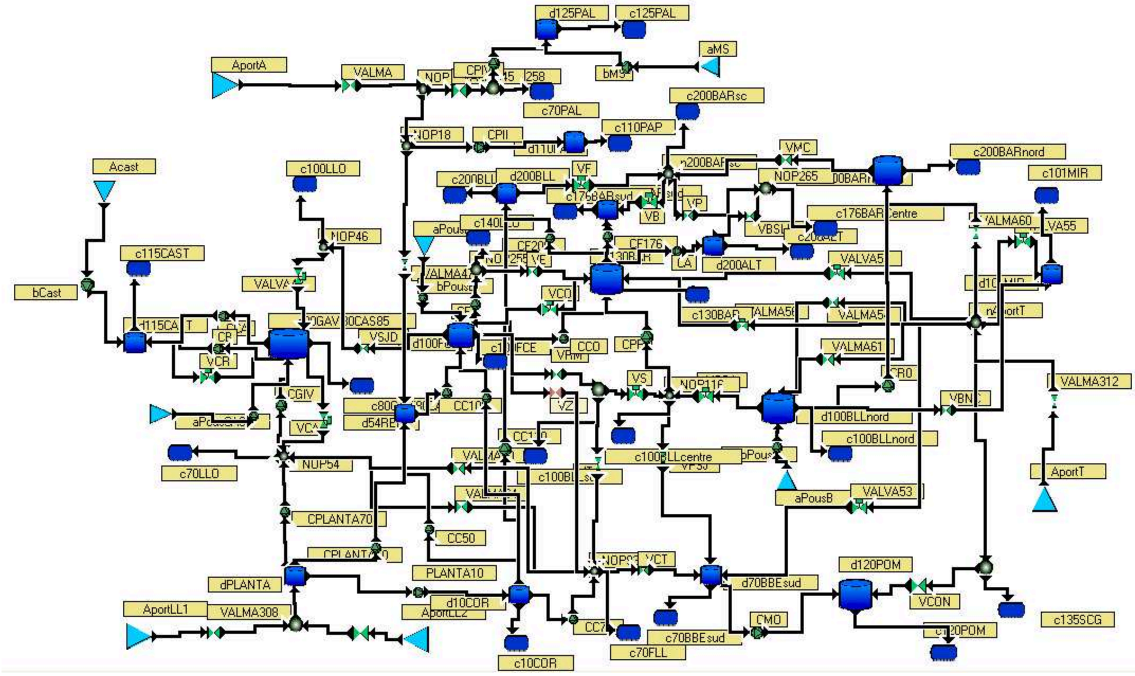


Figure 8: *Aggregated Barcelona network model in PLIO tool*

some objectives like:

1. the satisfaction of water demands;
2. the preservation of security levels in the reservoirs;
3. the minimization of the supply, production and transport cost;
4. actuators (valves and pumps) should operated as smoothly as possible.

4.2 Parameters of Network Elements

In this section, there is the description of the parameters of the hydraulic elements.

4.2.1 Tanks

The *tanks* are state variables of the system and their water level represents the outputs of the system. In those elements, the minimum and the maximum water volume should be specified. In addition, a penalty volume can be selected, which is the minimum security volume that enables to satisfy the whole demand and also represents a reserve when there are some malfunctioning, like pipe broken or similar faults.
Table 1 shows the physical limits for every tank in the network. First of all the tank name is reported, that allows to identify it in the model of Figure 7. In Figure 7, for every tank as well as the name, the vector state variable used by controller is shown ($[x_1, \cdots, x_{17}]$) in the second column of the Table 1. In the last two columns, the maximal and the minimal volume values are indicated.

Tanks name	Vector state	Minimum volume $[m^3]$	Maximum volume $[m^3]$
d125PAL	x_1	150	445
d110PAP	x_2	375	960
d115CAST	x_3	198	3870
d80GAVi80CAS85CRO	x_4	480	3250
dPLANTA	x_5	0	14450
d54REL	x_6	800	3100
d100FCE	x_7	16500	65200
d10COR	x_8	0	11745
d200BLL	x_9	700	7300
d130BAR	x_{10}	3840	16000
d176BARsud	x_{11}	200	1035
d70BBEsud	x_{12}	22450	98041
d200ALT	x_{13}	500	4240
d100BLLnord	x_{14}	6000	37700
d200BARnord	x_{15}	700	7300
d101MIR	x_{16}	1403	4912
d120POM	x_{17}	150	1785

Table 1: *Tanks physical characteristics: the volume value is reported in cubic meters*

4.2.2 Pumps

The *pumps* are one type of the actuators (active elements) in the network. The pumps, presented into the model, are 26: among of them, 5 (*bMS, bCast, bPousCAST, bPousE, bPousB*) are associated to the underground sources and the others are used to carry the water where there is an different ground elevation between two different elements.
In case of pumps, two parameters should be specified: electrical costs for every hour and the maximum flow, which represents the maximum water's amount of water that the pump is able to pump out in one second, in m^3/s. The minimal flow is not mentioned because it is always zero.
For each pump, the maximum flows are reported in the Table 2. Moreover the position occupied by every pumps i in the input vector u used by the controller is presented in the second column:

$$[u_i \quad \text{with } i = 1, \cdots, 61]$$

and in the first column, as it is in the case of tanks, the name.
The other parameters that have to be considered for the pumps are the electrical costs. These parameters play a fundamental role in the computation of the production cost (FCP), the part of the cost function which depends on the economical price.
These costs change with the daily hours and they have some different behaviours.
The first kind of pumps cost is the most common one: it can be divided into five time slots, as it can be seen in the Table 3.
Analysing the Table 3, it can be noticed that these pumps have the same cost in the night from 00.00 a.m. to 7.00 a.m. After this time, it increases in the morning from 8.00 a.m. to 15.00 p.m., and even more in the afternoon from 16.00 p.m to 21.00 p.m. At this time, the costs slowly decrease at 22.00 p.m. and something more at 23.00 p.m.
The second type of pumps cost is characterised by 4 time slots.

These pumps have a fixed price in the night from 00.00 a.m. to 08.00 a.m., which increases in

[2]in italics there are the pumps with assumed maximal flow values, it was necessary for the lack of the real data

Pumps name	Input vector	Maximum value $[m^3/s]$	Pumps name	Input vector	Maximum value $[m^3/s]$
CPIV	u_3	0,0317	bMS	u_4	0,0150
CPII	u_5	0,0220	bCast	u_7	10^{-5}
bPousCAST	u_9	0,0056	CCA	u_{10}	0,1200
CB	u_{11}	0,0500	CPLANTA70	u_{15}	0,2900
bPousE	u_{17}	0,2300	CGIV	u_{19}	0,0108
CPLANTA50	u_{20}	1,8000	PLANTA10	u_{21}	2,9000
CE	u_{22}	0,6200	CRE	u_{23}	3,0000
CC100	u_{24}	3,1000	CC50	u_{27}	0,6000
CF200	u_{29}	0,2600	CC130	u_{33}	0,0900
CC70	u_{33}	0,4000	CF176	u_{36}	0,1563
CCO	u_{38}	0,8500	CA	u_{42}	0,4250
CPR	u_{45}	$0,005$	CMO	u_{48}	0,0250
CRO	u_{53}	0,1342	bPousB	u_{55}	0,3800

Table 2: *Pumps physical characteristics*[2]*, the maximal flow is in cubic meters per second*

Pumps name	Input vector	00 a.m. 07 a.m.	08 a.m. 15 p.m.	16 p.m. 21 p.m.	22 p.m.	23 p.m.
CPLANTA70	u_{15}	0,0184	0,0285	0,0330	0,0285	0,0281
CPLANTA50	u_{20}	0,0109	0,0168	0,0194	0,0168	0,0166
PLANTA10	u_{21}	0,0014	0,0021	0,0024	0,0021	0,0021
CE	u_{22}	0,0113	0,0179	0,0193	0,0179	0,0169
CRE	u_{23}	0,0177	0,0186	0,0202	0,0186	0,0177
CC100	u_{24}	0,0223	0,0345	0,0398	0,0345	0,0340
CC50	u_{27}	0,0050	0,0077	0,0089	0,0077	0,0076
CF200	u_{29}	0,0217	0,0342	0,0365	0,0342	0,0324
CC130	u_{33}	0,0546	0,0845	0,0978	0,0845	0,0835
CC70	u_{34}	0,0168	0,0260	0,0300	0,0260	0,0257
CF176	u_{36}	0,0157	0,0247	0,0264	0,0247	0,0234
CCO	u_{38}	0,0221	0,0346	0,0369	0,0346	0,0327
CA	u_{42}	0,0222	0,0353	0,0381	0,0353	0,0334

Table 3: *First type of pumps electrical costs, the costs are in euro per cubic meters*

Pumps name	Input vector	00 a.m. 08 a.m.	09 a.m. 18 p.m.	19 p.m. 22 p.m.	23 p.m.
CCA	u_{10}	0,0174	0,0230	0,0265	0,0230
CGIV	u_{19}	0,0590	0,0779	0,0900	0,0779
CR0	u_{53}	0,0487	0,0644	0,0744	0,0644

Table 4: *Second type of pumps electrical costs, the costs are in euro per cubic meters*

the middle of the day from 09.00 a.m. to 18.00 p.m. and even more also in the evening from 19.00 p.m. to 22.00 p.m. At 23.00 p.m. the cost slightly decreases. This behaviour is reported in Table 4.

In the current version of the model, the pumps related to the underground sources are not considered, that it, there are still five pumps which it is necessary to assign an electrical cost. Three of these (*CPII, CPIV* and *CB*) have each one a particular cost type, as it is shown in the Tables 5, 6 and 7. The last two pumps, *CMO* and *CPR*, in practise are only used in some emergency cases. So, this in reflected in the optimization problem by imposing a very big constant cost, in order to the controller use them not much.

Pumps name	Input vector	00.00 a.m. 23.00 p.m.
CPII	u_5	0,0003

Table 5: *CPII pump electrical costs, the costs are in euro per cubic meters*

Finally, the economical cost trend of the pumps *CPII, CPIV* and *CB* is given:

- CPII pump has a fixed cost for every hour in the day according to the Table 5.
- CPIV pump has a three time slot cost: the first, the most long, is from 00.00 a.m. to 18.00 p.m., at 19.00 p.m. the cost increases and it remains the same until 22.00 p.m. At 23.00 p.m., this cost decreases to the first value.
- CB pumps cost could be divided into four time slots, although in the first two periods the same value appears. It is due to an approximation, because the difference between these two values is very small. These prices are shown in Table 7.

Pumps name	Input vector	00.00 a.m. 18.00 p.m.	19.00 p.m. 22.00 p.m.	23.00 p.m.
CPIV	u_3	0,0003	0,0005	0,003

Table 6: *CPIV pump electrical costs, the costs are in euro per cubic meters*

Pumps name	Input vector	00.00 a.m. 08.00 p.m.	09.00 a.m. 18.00 p.m.	19.00 a.m. 22.00 p.m.	23.00 p.m.
CB	u_{11}	0,0003	0,0003	0,0009	0,003

Table 7: *CB pump electrical costs, the costs are in euro per cubic meters*

Additionally to the electrical cost, the price of the water in the source should be considered.

4.2.3 Valves

In this model there are 35 valves. These elements are also active elements in the network, but unlike pumps, they do not have an electrical cost. In fact, they are not able to drive the water from a hydraulic element to another with different elevation. The valves can only let the water pass through or not, and to establish the flow path. It is important to notice that there is always a valve after an external source, in order to decide the amount of water that it can be injected to the network.
In PLIO tool, the parameters of the valves, which have to be set, are the maximal flows (Table 9).

Valves name	Input vector	Maximum value $[m^3/s]$	Valves	Input vector	Maximum value $[m^3/s]$
VALVA	u_1	*1,297*[1]	VALVA45	u_2	0,05
VALVA47	u_6	1,2	VCR	u_8	0,03
VALVA308	u_{12}	*5,34*[3]	VALVA48	u_{13}	0,22
VCA	u_{14}	0,065	VALVA309	u_{16}	*2,5*[3]
VSJD	u_{18}	0,75	VALVA64	u_{25}	no limits[4]
VALVA50	u_{26}	0,1594	VF	u_{28}	0,29
VE	u_{30}	0,45	VRM	u_{31}	3,5
VZF	u_{32}	0,35	VB	u_{35}	0,15
VCO	u_{37}	0,5249	VS	u_{39}	1,2
VT	u_{40}	1,3	VCT	u_{41}	1,2
VP	u_{43}	0,15	VBSLL	u_{44}	0,15
VCOA	u_{46}	1,35	VPSJ	u_{47}	0,55
VMC	u_{49}	0,24	VALVA60	u_{50}	no limits[4]
VALVA56	u_{51}	1,7	VALVA57	u_{52}	0,4051
VBNC	u_{54}	0,392	VALVA53	u_{56}	1,5001
VALVA54	u_{57}	1,7361	VALVA61	u_{58}	no limits[4]
VALVA55	u_{59}	0,1852	VCON	u_{60}	0,035
VALVA312	u_{61}	*6,27684*[3]			

Table 9: *Valve maximal flows*

In the Table 9, as for the tanks and the pumps also, the name and the order in the input vector u is reported.

In addition, it is possible to stabilise these element, by including them into the stability function. Stability could be enforced for every valve, and even to enforce it more in some valves than other by specifying particular weights. This is the case, for example, in valves controlling external sources that come from potabilization plants that can not be start and stop continuously.
Some valves have the minimum value different from zero, but, for the moment they are fixed to zero. These minimal values different from zero are due to the fact that, sometimes, it is impossible to close totally a valve, but a small amount of water continues to pass through it.

4.2.4 Sectors of Consume

The sector of consume represents the demands of users. It is considered like a known disturbance. The pattern demands used are provided by AGBAR. The demand patterns reflect the real profile of the city consume during the 24 hour of the day. For the moment every day has the same profile, but, in the future, it would be possible have a different pattern according the days in the week.

In the Figure 9, it is possible to notice that the profiles of the demand sectors are equal in each of the three tanks, but the range is different since it has been scaled according to the total consume of the sector in a day. In the future, the particular demand profiles of each sector will be used.
The model, as it has been already explained, is an aggregation of the real network. The demand

[3]this values in italics, in the reality, are non limits but in the model they follow the external sources and so they get the max water flow supplied by each sources

[4]This valves with a non limit flow, in the model, are set to $15m^3/s$, a very high value

Sector name	Vector demands	Maximum request $[m^3/s]$	Minimum request $[m^3/s]$
c125PAL	d_1	0,015	0,004
c70PAL	d_2	0,009	0,003
c110PAP	d_3	0,018	0,005
d115CAST	d_4	0,018	0,005
c100LLO	d_5	0,631	0,203
c80GAVi80CAS85CRO	d_6	0,194	0,062
c70LLO	d_7	0,264	0,085
c200BLL	d_8	0,023	0,007
c140LLO	d_9	0,216	0,069
c10COR	d_{10}	0	0
c176BARsud	d_{11}	0,270	0,087
d130BAR	d_{12}	1,262	0,406
c100FCE	d_{13}	0,445	0,143
c100BLLsud	d_{14}	0,224	0,072
c70FLL	d_{15}	1,455	0,468
c200BARsc	d_{16}	0,411	0,132
c100BLLcentre	d_{17}	0,478	0,154
c70BBEsud	d_{18}	2,908	0,936
c200ALT	d_{19}	0,137	0,044
c176BARcentre	d_{20}	0,137	0,044
c100BLLnord	d_{21}	0,575	0,185
c120POM	d_{22}	0,014	0,004
c200BARnord	d_{23}	0	0
c101MIR	d_{24}	0,631	0,203
c135SCG	d_{23}	0,018	0,060

Table 10: *Tanks physical characteristics*

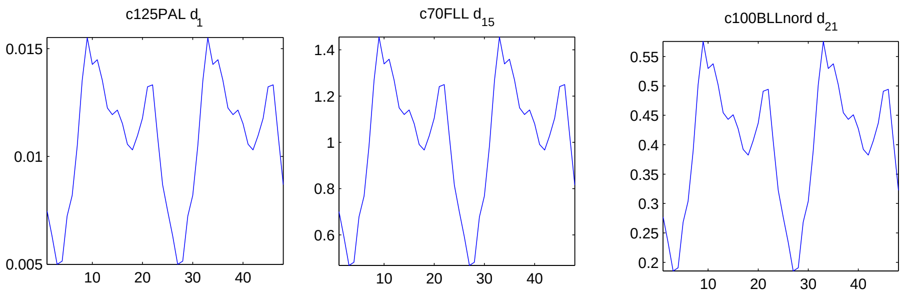


Figure 9: *The profiles of some demand sectors*

values are obtained through the calculation of this aggregation process.
In the aggregated network, 25 demand sectors appear. In the Table 10 the name of the sector, the order in the demand vector and, finally, the operational range (maximal and minimum values) are reported. The first thing that it can be noticed is the presence of two sector with the maximal and minimal demand equal to zero. This is due to that these values results from a statistical analysis and not from a analysis instant by instant. Moreover, there is extrapolation process of a fixed value per second every hour.

4.2.5 Sources

Source name	Type	Correspondent actuator	Maximal contribution
AportA	VALVA	superficial	1,297
aMS	bMS	underground	0,0150
aCast	bCast	underground	0,0056
aPousCast	bPousCast	underground	10^{-5}
AportLL1	VALVA308	superficial	5,34
AportLL2	VALVA309	superficial	2,5
AportT	VALVA312	superficial	6,27684
aPousE	bPousE	underground	0,23
aPousB	bPousb	underground	0,38

Table 11: *Source maximal contribution*

The sources present in the model are nine, four external and five underground. The external sources are modelled using a source element and a valve, while the underground sources involve a source element followed by a pump, which is necessary for the water extraction from the underground.
Comparing the Table 11 with the Tables 2 and 9, it is possible to notice that the maximal contribution of every source is equal to the maximal flow in the actuator, as it is explained before.

5 Construction of a Water Network Simulation Environment

This chapter provides a detailed description of the water network simulation environment develop using Simulink® /Matlab® tool. As an application, it has been applied to the Barcelona network.

Simulink® is an environment for multi-domain simulation and model-based design of dynamic and embedded control systems. It provides an interactive graphical environment and a customizable set of block libraries that allow to design, simulate, implement, and test a variety of systems, used in communications, control, signal processing, video processing, and image processing. Thus, Simulink® represents the appropriate tool to develop a water network simulation environment that allows to include both a network model and the cost function computation. This simulation environment allows to interface with the controller, by the moment developed in Matlab® or in PLIO, which are able to provide the set-points for the actuators. The Simulink® structure has been developed to obtain a tool easy to handle and where it is possible to change the parameters and the cost function formulation in a simple way. In the future, the Simulink® model could be connected to a whatever controller, and in a second stage, to close the feedback control loop. The final purpose of this simulator is the evaluation of the performances of the controller and the comparison of different controllers.

5.1 The Simulation Environment

The simulator, at the beginning, need the parameters of every element and the value of the actuator set-point or the demands. All this data, when generated using PLIO tool, is saved in an *Microsoft Access*[6] database. In this database, there are the values of each element computed at every iteration. In Matlab®, it is possible to load values from the database, using a specific toolbox, the *Database Toolbox*™.
The *Database Toolbox* ™ is, indeed, a product that provides a tool to exchange data between Matlab® and any *ODBC/JDBC-compliant* database. With the Visual Query Builder tool within the toolbox, stored data can be querier without needing to use SQL. This gives the ability to access, analyze, and store data quickly and easily from within Matlab®.
It has been decided that the data loaded from the database in the workspace are saved into a different structure, for every different element. For simplicity these structures have the same name of every element in the network. The number of fields are different depending on the element type. The structure is only created for the element for which it is necessary to set some parameters, as for the tanks, the pumps or the valves.
When the Simulink® model is connected directly to a controller developed in Matlab®, otherwise, the values of the simulation results are stored in the workspace, and it is sufficient to insert them in the corresponding data structures.

Figure 11 present the main window of the water simulator environment. The blocks showed in Figure 11 provide a tool to load all the data necessary to parametrize every element in the model.
According to the origin of the data, there are two different blocks: one to load the data from the database (in blue), in the case of PLIO, the other to load the data from a *mat* file (in pink). Indeed, it is preferable, after extracting the data from the database, to save the data in a *mat* file in order to save computation time, since the access to the database is quite slow.

[6]Microsoft Access is a relational database management system from *Microsoft* that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools.

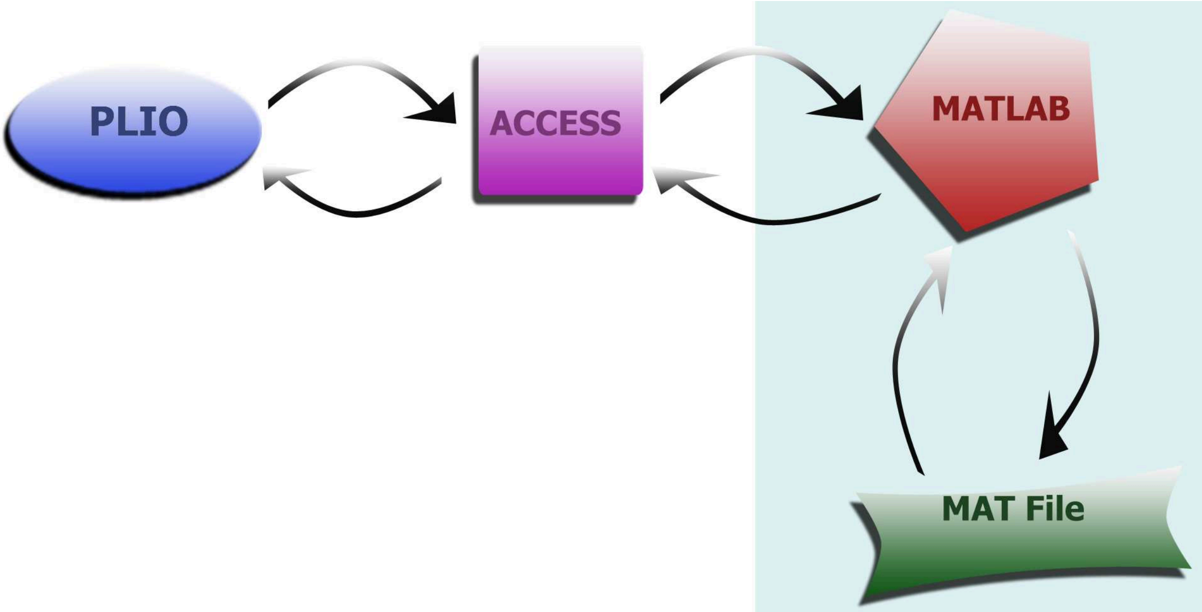


Figure 10: *The connection between the different environment used which allows to interface us*

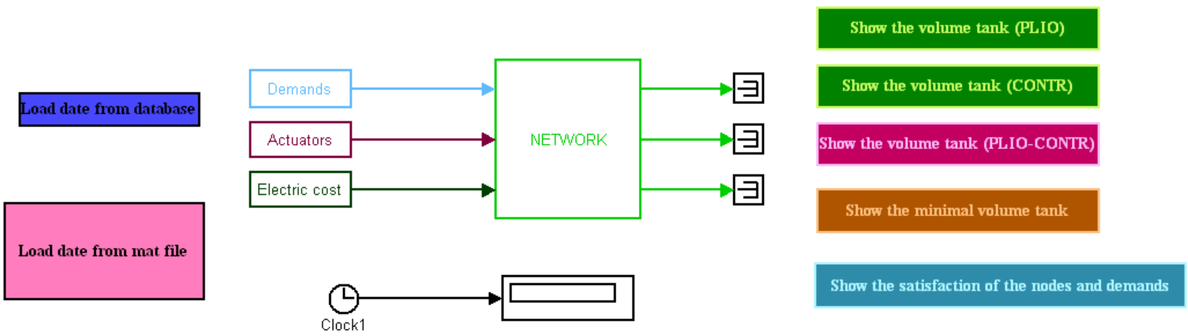


Figure 11: *The main window of* SIMULINK® *model*

Notice from the Figure 11 that the model is used in open loop. In fact, the inputs are represented by the three block called demands, actuators and electrical cost and they are not in a feedback loop. This means that the controller does not use the results of the simulation, at each iteration, but the SIMULINK® model is useful for a off-line checking. When the simulator would be closed loop with the controller, it would provide the value of the actuators at every sample time, then the loop would substitute the actuators input.
The model for a water network is built using a different block for every different element. Figure 12 shows, the different colours used for every element:

- the *pink* for the tanks;
- the *yellow* for the sources;
- the *green* for the pumps;
- the *blue* for the valves;
- the *sky blue* for the demand sectors;

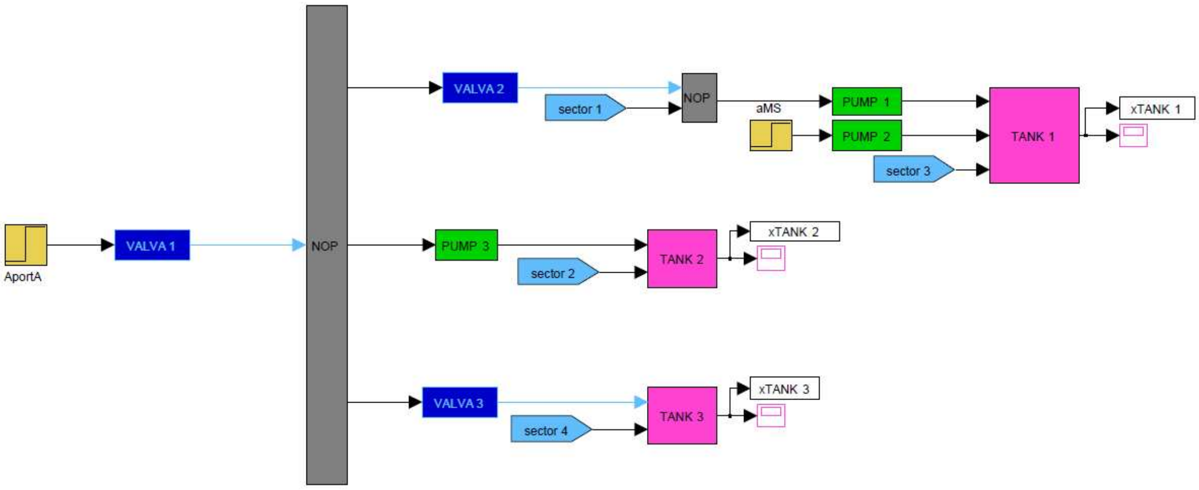


Figure 12: *An example of a small network model*

- the *grey* for the nodes.

Figure 12 is an simple example of a network where the different type of elements are presented. In order to manage all the connections between the network inputs and outputs, the input/output block in Figure 13 has been generated.

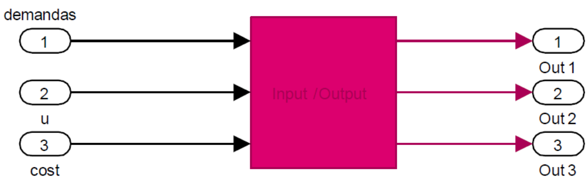


Figure 13: *The input/output block in the network*

This block is at the same level than the network, inside the green NETWORK block in the Figure 11. Under this block there is a very complex subnetwork, which considers the necessary inputs, or outputs of the network. In addition, in this network there is the computation of the cost function. In the following, a detailed description of the strategy used to implement every element type is reported. Moreover, how the input-output block computes the cost function is also explained .

5.2 Element Analysis and Implementation

Every element has a particular implementation, based on a sub-network which allows to reproduce its dynamical behaviour. Each element is always implemented using the same structure. The differences, that could appear in two subnetworks of the same element type, are due, only, to the output or input number in tanks or nodes model blocks.
The actuators, otherwise, do not have differences in the number of inputs or outputs, since they have always only one input and one output. They present a different structure when they are behind a source.

In addition, for every element, a structure in which stores all its parameters is created in the workspace. This structure has a different number of fields for each different type of element.

5.2.1 Sources

The sources are the most simple type of element in the network for the SIMULINK® implementation. Every source is represented with a step, where its value is the max flow allowed. The source flows are controlled by means of the actuators which follow the sources. These elements are implemented using the block step directly coming from the SIMULINK® library.

5.2.2 Actuators

The actuator block models have only one input and one output, and take these values, directly, from the workspace. Their implementation aims, only, to guarantee that the value, computed outside the simulator in the controller, are into the range of the particular element. This is assured by a saturation block that has as upper bound the maximal flow and as lower bound the minimal flow allowed to pass through the actuator. When the element flow is out of its operating range, the saturation block assigns the upper or the lower bound value depending if the flow is bigger that the maximal or smaller than the minimal.
In the actuator implementation there is, also, the determination of the term used to compute the objectives in the cost function. In particular, for the actuators, it is important the stability term. Moreover, for the pumps, the economical term is computed such that it reflects the electrical hourly consume. This is the main difference between valves and pumps.
When an actuator follows a source, the actuator structure need some modifications. In this particular case, is highlighted by the presence, into the subnetwork, of the block which loads the data from the workspace. In the other case, otherwise, it is presented in the subnetwork of the element which precedes the actuators.

Valves The valves are used to manage the flow of the water passing through. The structure created for the majority of the valves, has three fields:

1. *data*: the vector of the flow values with a dimension equals to the simulation horizon: there is a value for every sampling time;

2. *flow_max*: the maximal flow allowed in the valve;

3. *flow_min*: the minimal flow allowed in the valve;

This structure is valid not for all the valves because for the valves which follow a source it is necessary to add another field in the structure: the *cost*. This cost is due to the price of the water at every source, and it is the same for every hour in a day.
In the implementation of this element in SIMULINK® environment, a mask is created where it is possible to set the maximal and the minimal flows. This could be useful for a future automatisation of the model creation, where it would be possible to generate the network from a script file.
At the moment, the minimal flow is always zero, but, in the future studies, it would be necessary to set a different value, since in pratice sometimes it is not possible to close completely the valves.
Under the mask for every valve, it is found a subnetwork which, as it is explained above, mainly,

simply checks if the values in the workspace are into the its operational range or not.
There are two different types of networks which implements a valve element: one is used when a valve follows a source, the other in all others cases. The differences between these two schemes are, in this case two: the first consists in the presence of a term which comprises the economical cost; the other in the place where the values are loaded form the workspace.
In the valve, used after a source, this is done in the own sub-network of the valve, while in the other case it is implemented in the element which comes before the valve, and the value obtained is sent to the valve block as a input.
In the Figure 14 the implementation of a valve which follows a external source is shown .
The switch block at the beginning is used to select the input. Indeed it is possible to choose between the source output flow or the flow values saved in the workspace. This option would be useful for future studies. In fact, it could be necessary to modify the source implementation which would have different values from the maximal and would regulate the output flow alone. At this time, the different values are obtained trough the valve, so the switching block always select the value from the workspace.
In addition, the structure in Figure 14 shows another branch which is used to compute the economical objective in the cost function. There is a multiplication between the flow and the cost, instant by instant. The meaning and the implementation of this term is explained in detail in the next Section 5.2.3, because it is equal to the electrical cost of the pumps.
The other part of the subnetwork appears in both actuator types, as it is possible to see comparing the Figure 14 and 15, regarding the valves and the Figure 16 and 17, regarding the pumps, and it is used to compute the stability term.
This term is obtained as:

$$\text{VALVA\_stab} = [u_1(t) - u_1(t-1)]^2$$

where the value $u_1(t-1)$ is obtained using the memory block, and the square multiplying the difference by itself. This term penalizes the differences between the flow value of the valve in two consecutive sample times. This value is passed to the input-output block where the total

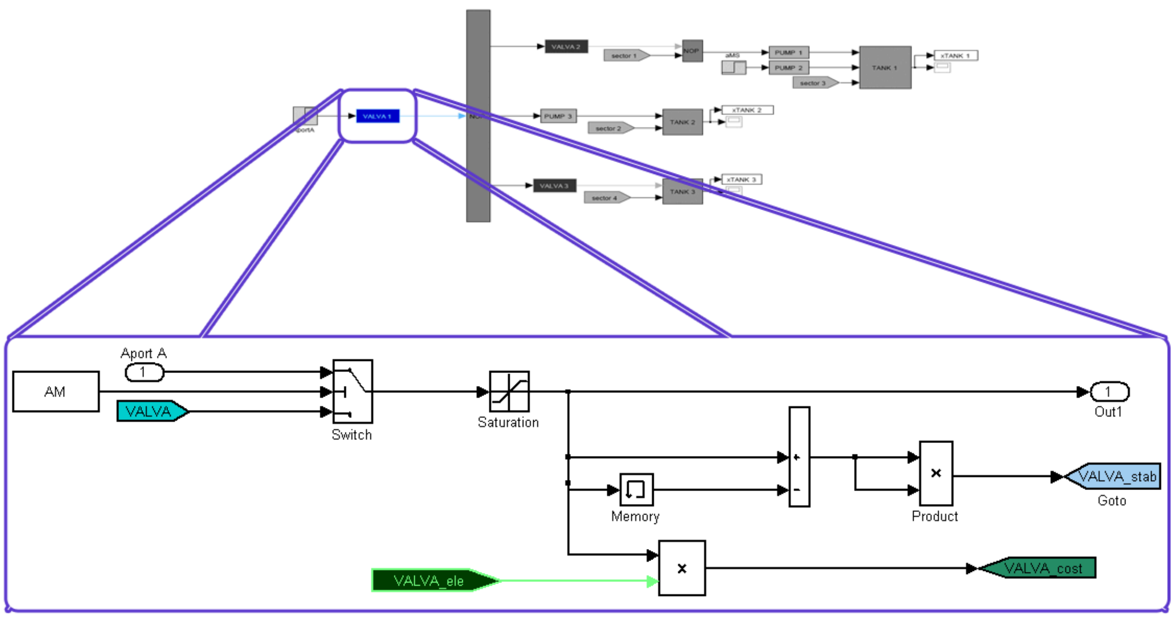


Figure 14: *Implementation structure of a valve which follows a source*

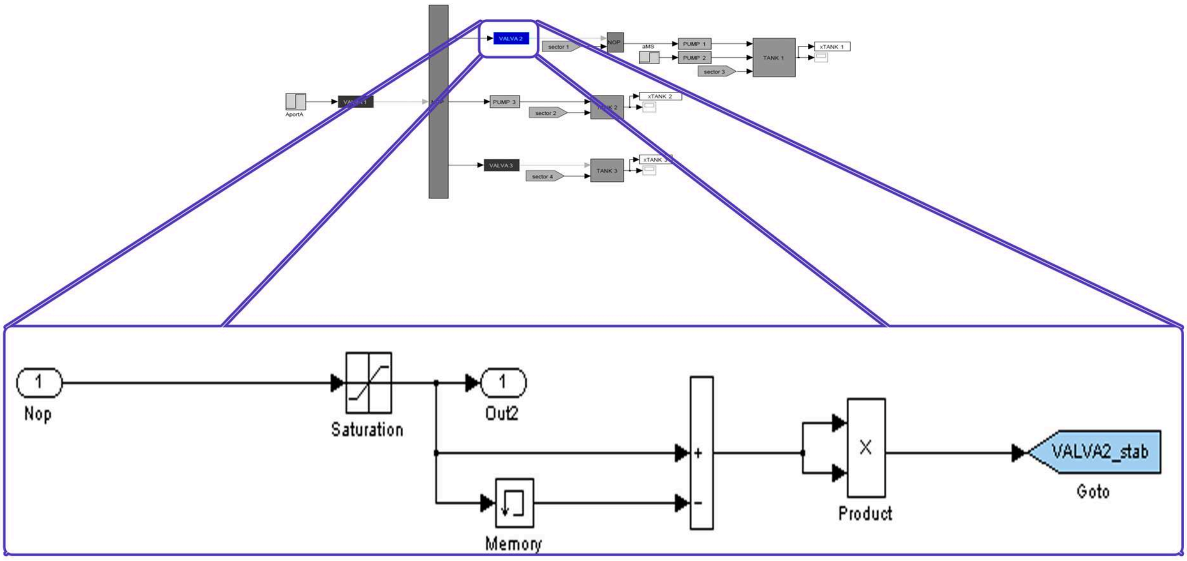


Figure 15: *Implementation structure of a "normal" valve*

cost function is computed, where all the coefficients for every actuator are considered.

5.2.3 Pumps

The pumps, the other type of actuators in the network, have a structure in the SIMULINK® model very similar to the valve. The difference in the implementations of these two actuator type in the network is that the pumps are able to drive up the water from a ground elevation to an bigger other unlike the valves. In general, however, the role of both types of elements is the same. They could decide the amount of water that could pass through by means of the local controller set-point established by the MPC controller. During the optimization, the best value for the actuators is selected according to the objective function.
The structure created in the workspace for every pump includes four fields. The first three are equal to those explained in the "normal" valve while the last one depends on the economical cost, as it was shown in the valve used after a source:

- *data*: the vector of the flow values, with a dimension equal to the simulation horizon;
- *flow_max*: the maximal flow allowed in the pump;
- *flow_min*: the minimal flow allowed in the pump;
- *cost*: the vector of economical costs that in this case the dimension of the vector is also to equal to the simulation horizon: there is a value for every time instant. The cost changes according to the hourly fare.

The difference between the cost in the pump and in the particular valve is due to the electrical cost of the pump changes hourly, while the water cost is considered always the same for every hour in the day.
As in the valve, there are two types of implementation: first type is used only when the pumps follow an underground source. Figure 16 shows the structure of a pump used after a source.
The second type of implementation is represented in Figure 17, where it could be noticed that

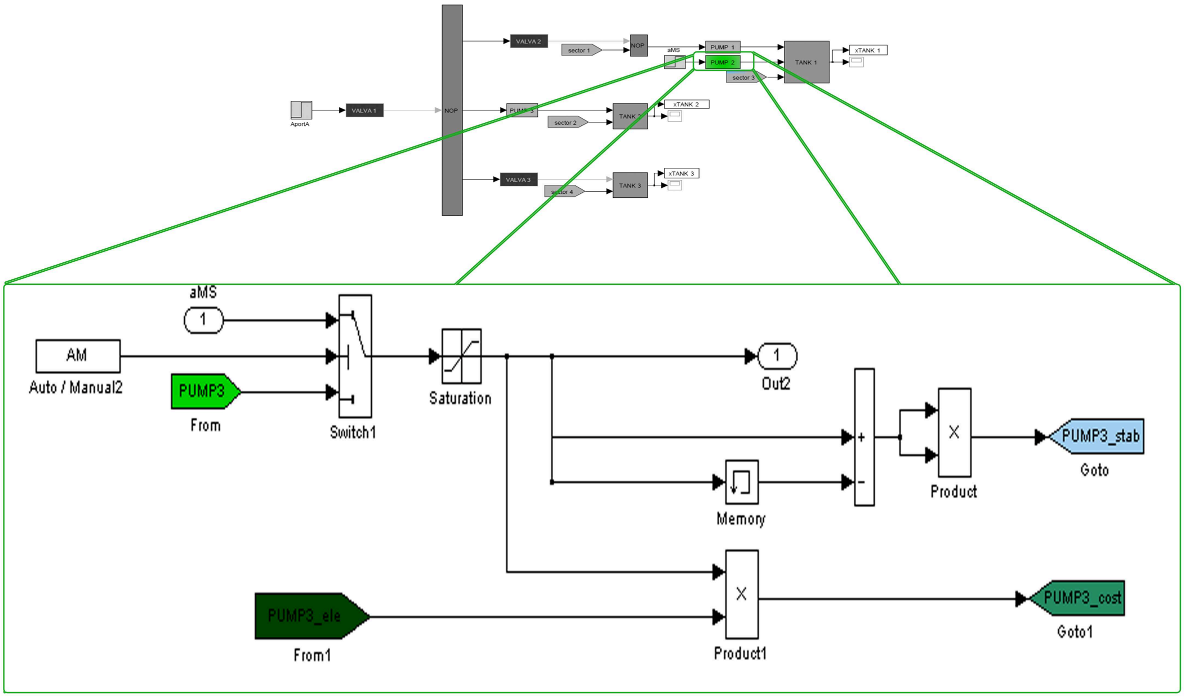


Figure 16: *Implementation structure of a pump which follows a source*

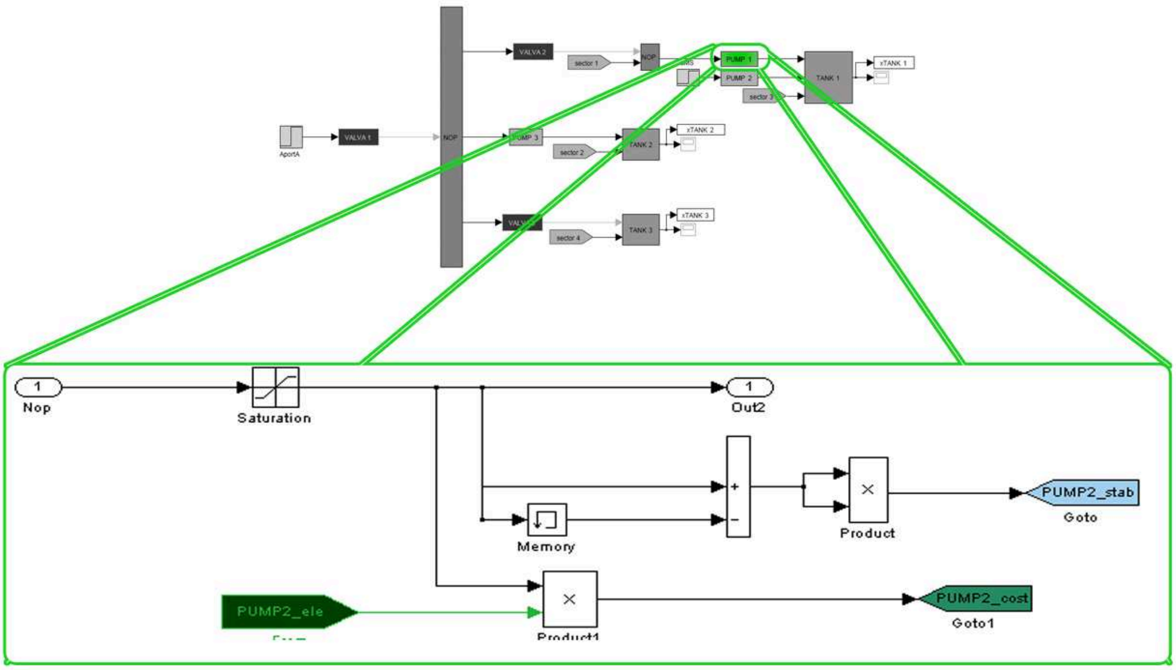


Figure 17: *Implementation structure of a pump*

the principal structure in Figure 16 is the same of that in Figure 14, as well as the structure shown in Figure 17 is equals to that in Figure 15. So, for these parts of the subnetworks, it is possible to do the same consideration made above, (see Section 5.2.2 corresponding to the valve

description).
However, in both pumps structures implementation, as in the valve which follows a source, there is a branch where the economical cost coefficient is computed. This represents the only difference between "normal" valves and pumps.
The hourly economical cost is loaded from the workspace in the main windows (Figure 11) and through the input/output block is sent as input to every corresponding pump, or valve. Considering the example in Figure 17, the cost in input is represented by the variable PUMP2_ele, in dark green.
The hourly cost has to be multiplied, instant by instant, for the flow in order to obtain the coefficient used in the cost function. These values are saved in the variable PUMP2_cost and passed to the input/output block where the total cost function is computed, considering the influence of every pump and valve.

5.2.4 Nodes

Dynamical models of the nodes are implemented as constraints, where the mass balance must be satisfied: that is the sum of the input flow has to be equal to the sum of output flow. The values computed by the controller should satisfy this constraint. But, in the simulator, it is decided to check this balance and give a signal when it is not respected. In this way, the simulator becomes, also, a tool which is able to evaluate the controller operation.
These elements, since are only constraints, do not need a structure, since they do not have particular parameters which have to be set. The difference between a node to an another consists only of the different number of inputs and outputs.

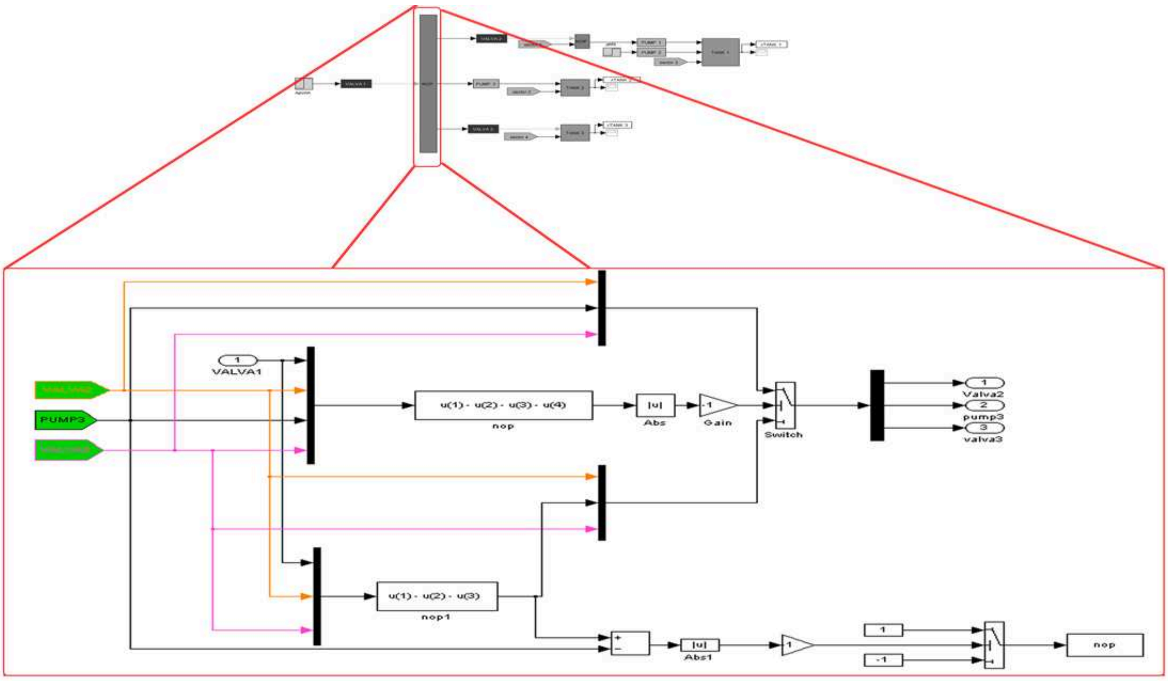


Figure 18: *Node implementation structure*

In Figure 18 the model implementation of a node with one input and three output is reported. So, the equation implemented in this node is:

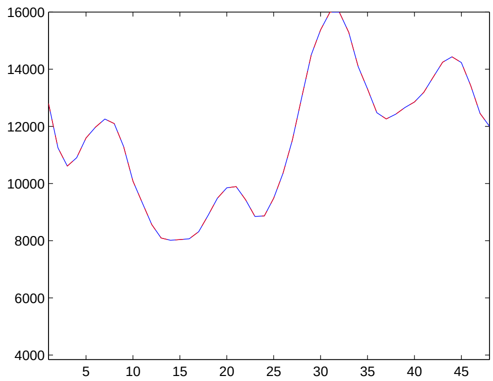
$$u(1) - u(2) - u(3) - u(4) = 0 \tag{20}$$

The mass balance of the flow in input and the flow in output has to be respected at every time. In fact the $u(i)$ represents the instantaneous flow in the actuator i. The output in the sub-network in Figure 18 is shown with the green block. These blocks implement the loading from the workspace. The equation (20) is the same of the constraint imposed in the controller. The simulator checks if the constraint is satisfied and in reverse it re-computes the balance changing the value of one output. This action is done in the part of network that is downstream the equation block. Here, there is a switch that checks, using a threshold, if the balance is satisfied or not. The balance is satisfied when this equation is about zero. When the constraint is not respected, one output has been considered like a free variable, and, instead of using the value in the workspace, as output, the value computed in the second equation block is used. There are another two branches in the structure implemented that aim to check the satisfaction of the constraint and the demand. The block with the name of the node generates a variable in the workspace which has as many ones as time instants in which the node balance is satisfied. When this constraint is not respected appears a minus one. Coming back to the main window of the model (Figure 11), there is a block which allows to visualize these variable and searches the minus ones. This block is on the right in sky-blue. When a minus one is found, in the command window of MATLAB® is displayed the number of the node and the corresponding instant when the balance is not satisfied.
The other part of the scheme is used to check if the demand, connected to this node, is satisfied at every instant or not. In fact, the sum in the lowest part computes the balance of the node without the demand, and, only after, the demand is subtracted. If the results of this difference is about zero the demand is satisfied, otherwise is not. The pale blue blocks in the Figure 24 generate the variable which have to be sent to the input/output block, appearing a minus one when there is a infeasibility, in the same way as the node balance.

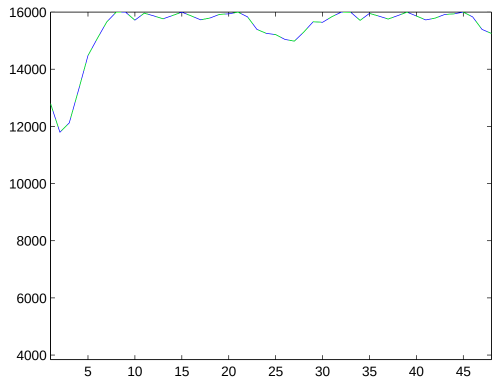
5.2.5 Tanks

The tanks are the dynamical elements in the system and they play a fundamental role in the simulator. Indeed, known the inputs and the disturbances (the demands), the tank volumes are calculated instant by instant.
First of all, it is important to check if the behaviour of the tank volume computed in this way is equal to that computed in the controller. When there are no infeasibilities, which could modify some actuator values, the results from the controller and the simulator are exactly the same.
These results are showed in Figures 19, where the volume behaviour of a tank in a simulation of 48 hours is displayed.
The two lines, in both figures, are completely overlapped, indicating that the SIMULINK® simulator allows to obtain the same results of those obtained both in the PLIO tool and in MATLAB® controller.
The structure of a typical tank implementation using SIMULINK® blocks is shown in Figure 20.

Figure 20 shows the implementation of a tank. In fact, this tank has only one input and one output. This structure could be more complicated when a tank has a bigger number of inputs or outputs.
Observing the network in the Figure 12, it can be noticed that the tank has three inputs, one corresponding to a demand. This is a trick to use the demand like a known disturbance and so it is necessary to load this data from the workspace. The demand is introduced in the equation



(a) *Volume tank behaviour in the simulator and in PLIO*



(b) *Volume tank behaviour in the simulator and in* MATLAB® *controller*

Figure 19: *The volume behaviour of a tank in a simulation of 48 hours: in the blue line the volume computed in the simulator while in (a) in the broken red line is displayed the volume resultant after the PLIO simulation, and in (b) in the broken green line is displayed the volume resultant after the* MATLAB® *controller simulation.*

as follows:

$$u(1) + u(2) - u(3) \tag{21}$$

where the second input $u(3)$ corresponds to the demand. The numbers into the round bracket indicate the input order in the multiplexer.
Multiplexer, equation and integrator SIMULINK® blocks implement the equation of the tank, that is, in this example:

$$V(t+1) = V(t) + \Delta t\big(u(t) - d(t)\big) \tag{22}$$

where $u(t)$ represents the sum of the inputs which, in this case, came from two pumps and the demand at time t. Comparing the equations (21) and (22), it can be noticed that in the first there is not the multiplication by Δt but this operation is done through the integrator, that plays a fundamental role in the scheme. In fact, after the equation block is obtained the instantaneous value of flow (in m^3/s). Passing this value through the integrator, the total flow in one hour is obtained. In addition, the integrator keep also in consideration the other difference between the equation (21) and (22): the volume at time before $V(t)$. Indeed the integrator at the first iteration adds to the equation block result the initial volume, and in the next iteration the volume at the time before.
The integrator needs to be set with the maximal capacity of the tank as upper bound and the minimal as lower bound. The initial condition, otherwise, is the initial volume of the tank.
The strategy used to implement the outputs is the same applied to the nodes, that is, the values are directly loaded from the workspace through the green blocks. In this case, otherwise, there is not a check, and these values are sent to the network as tank outputs without any changes.
The value of the volume calculated for every tank is saved in the variable with the tank name, the magenta block, which is a connection to the input/output block.
All tank volume behaviours are visualized through the green blocks on the right in the main

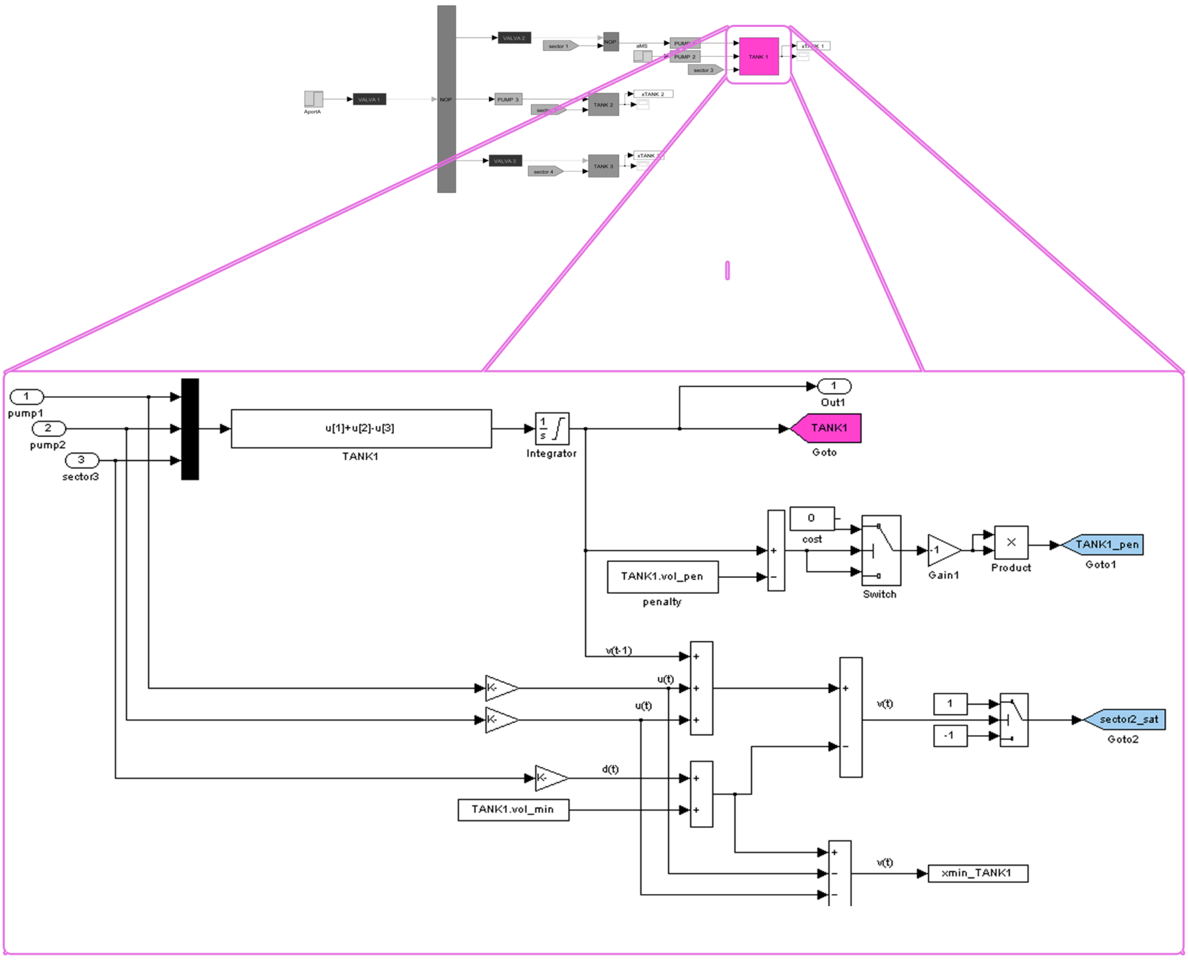


Figure 20: *Tank implementation structure*

windows (Figure 11). There are two different blocks: one used when the simulator runs with the value actuators computed with the PLIO simulation, the other when they are calculated by the controller simulation.

The magenta block in the main window, on the right, allows to compare the simulation results of PLIO tool or Matlab® controller. This is useful to show the differences of these two controllers in the same scenario.

Coming back to the tank structure there are another three branches which have not been already explained.

The first generates the coefficient of the penalty objective for every tank. The equation implemented is in (6), where the penalty volume is subtracted to the volume at every time and when this difference is bigger than zero, there is a penalty coefficient. This value is squared and after, in the block input/output, the total penalty cost function, including all tanks component, is computed. This value, which has to be passed to the next computations, is saved in the sky-blue block.

The second branch is equal to a part of the node implementation, indeed it generates the vector that indicates the demand satisfaction. In tanks case, it is necessary to take in account more things than in node case. In particular, the check does not only consider the input/output balance but it has to consider the volume stored in the tank to see if there is enough water to satisfy the demand. Moreover, the tank has a minimum level under which it is not possible to go. The minimum volumes for all the tanks are reported in the Table 1. So, also this value has to be included in the computation of the demand satisfy. In detail, the scheme makes these operations at every time t for the tank i, which are obtained using (22):

$$V_i(t) + \Delta t\big(u_{i,in}(t) - u_{i,out}(t)\big) \geq \Delta t\, d(t) + V_{i,min} \tag{23}$$

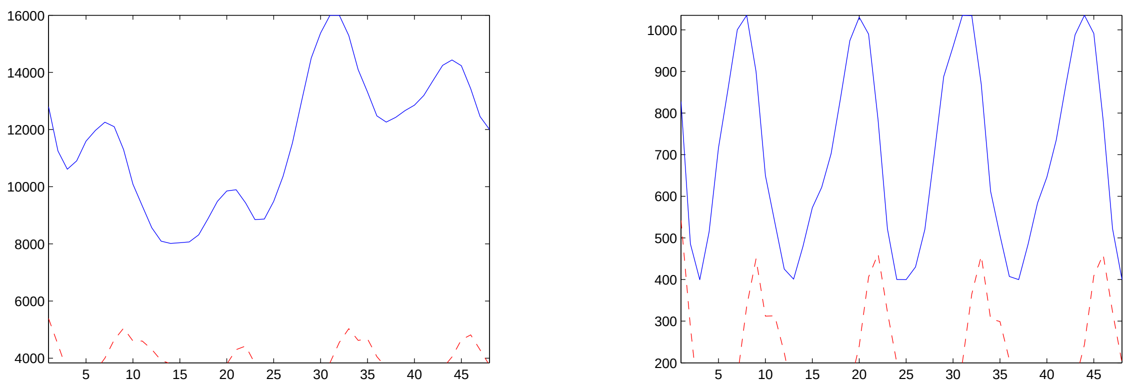
where first of all input and output flows in actuators are multiplied by Δt, the sampling time, in order to obtain the total amount of water in one hour. After this, the tank volume is computed without the demand, with a sum block (the part in (23) before the inequality signs). In order to satisfy the demand, this value has to be bigger or equal to the requested of water in this hour (the value of the demand is multiplies by Δt) plus $V_{i,min}$ the minimal volume of the tank i (the part in (23) after the inequality signs). To obtain a inequality respect to zero, the two term are subtracted, as follows:

$$V_i(t) + \Delta t\big(u_{i,in}(t) - u_{i,out}(t)\big) - \Delta t\, d(t) - V_{i,min} \geq 0 \tag{24}$$

As in the node, it appears a minus one in the variable used to validate the satisfaction of the demand when there is an infeasibility.
The remaining branch computes the minimal volume of the tank necessary, instant by instant, to satisfy the demand. This tool is very useful to discover the real penalty level of every tank. In fact, using this value there is not a lot waste of water, which is stored in the tank without being strictly necessary. This value is computed changing the signs in the equation (23) without considering the term $V(t)$. In fact, the volume needed to satisfy the demand in the tank i is $V_{i,min} + \Delta t\, d(t)$ where we have to sum the output and substract the input. That is:

$$\Gamma_i(t) = V_{i,min} + \Delta t\, d(t) - \Delta t\big(u_{i,in}(t) - u_{i,out}(t)\big) \tag{25}$$

The obtained values $\Gamma_i(t)$ are plot for every tank i together with the volume behaviour of the tank using the orange block in the main window. Figure 11 shows two tanks, as an example of usage of this tool.



(a) (b)

Figure 21: *Examples of the minimal volume computed in two tanks during a simulation of 48 hours*

When there is not an infeasibility, as in the case of in the Figure 21, the broken red line, which represents the minimal value computed, is always below than of the blue one, which indicates the stored volume of water in the tank during the simulation. As it is explained in the next chapters, many tests have to be done to identify the rigst penalty volume which allows to obtain the minimal cost.

The tank dynamics plays a main role in the analysis of the network behaviour. Indeed the comparisons between different controllers are done comparing the tank volumes.

5.3 Implementation of Barcelona Water Network Simulator

As an application of the library of elements built, a simulator for the aggregated network of Barcelona is developed (see Figure 22). Every element in the network in Figure 22 has the same name of the VISIO model presented in Figure 7, so it is quite simple to relate both models.
At the top left of Figure 11, there is a red block called input/output. This block takes care of all the connections between the network inputs/outputs and the element blocks. It is possible to notice that in Figure 23, the subnetwork under the input/output block is very complex. Every coloured element, in Figure 23, corresponds to a different variable in every element. In this block, the cost function is computed, as well as saved the matrix of system states and the matrix which indicates the satisfaction of demands.
Every block, in the network in Figure 22, respects the colour convention explained previously, as well as the subnetworks. The strategy used to create the dynamical model of every element is already explained, but it is interesting and useful to provide some details on how the implementation of the Barcelona aggregated model has been done. In particular, details about node and tank implementation, where the network has a great complexity due to the big number of inputs/outputs, are given.

In the Figure 24, the node *n*70FLL is shown. This node presents a very complex structure, due to a big number of inputs and outputs.
In particular, the node *n*70FLL (Figure 24) has 4 inputs and 3 outputs. One of the 3 outputs is a demand sector (the sector c70FLL), but, as in the tank, it appears as an input. The other outputs in Figure 24 are represented by the green block, which are the connections with the input/output block and so to the workspace. In this block there is the value saved in the workspace for every element, in the field *data*.
The equation of the mass balance, where the demand sector is considered as a negative input, is:

$$u(1) - u(2) + u(3) + u(4) + u(5) - u(6) - u(7)$$

where the number into the round brackets represents the order of the multiplex input. The demand sector is the second input in the multiplexer preceded by the minus sign.
Using the variable name used in the controller (Tables 2, 9 and 10), the constraint is:

$$u_{25} - d_{15} + u_{32} + u_{33} + u_{40} - u_{26} - u_{41} = 0 \tag{26}$$

This constraint is checked at every sample time, and it assures that there is not water stored in the nodes.

The Figure 25 shows the subnetwork regarding the tank d130BAR. The equation implemented in the tank d130BAR is very complex. This complexity is due to the number of inputs and outputs.

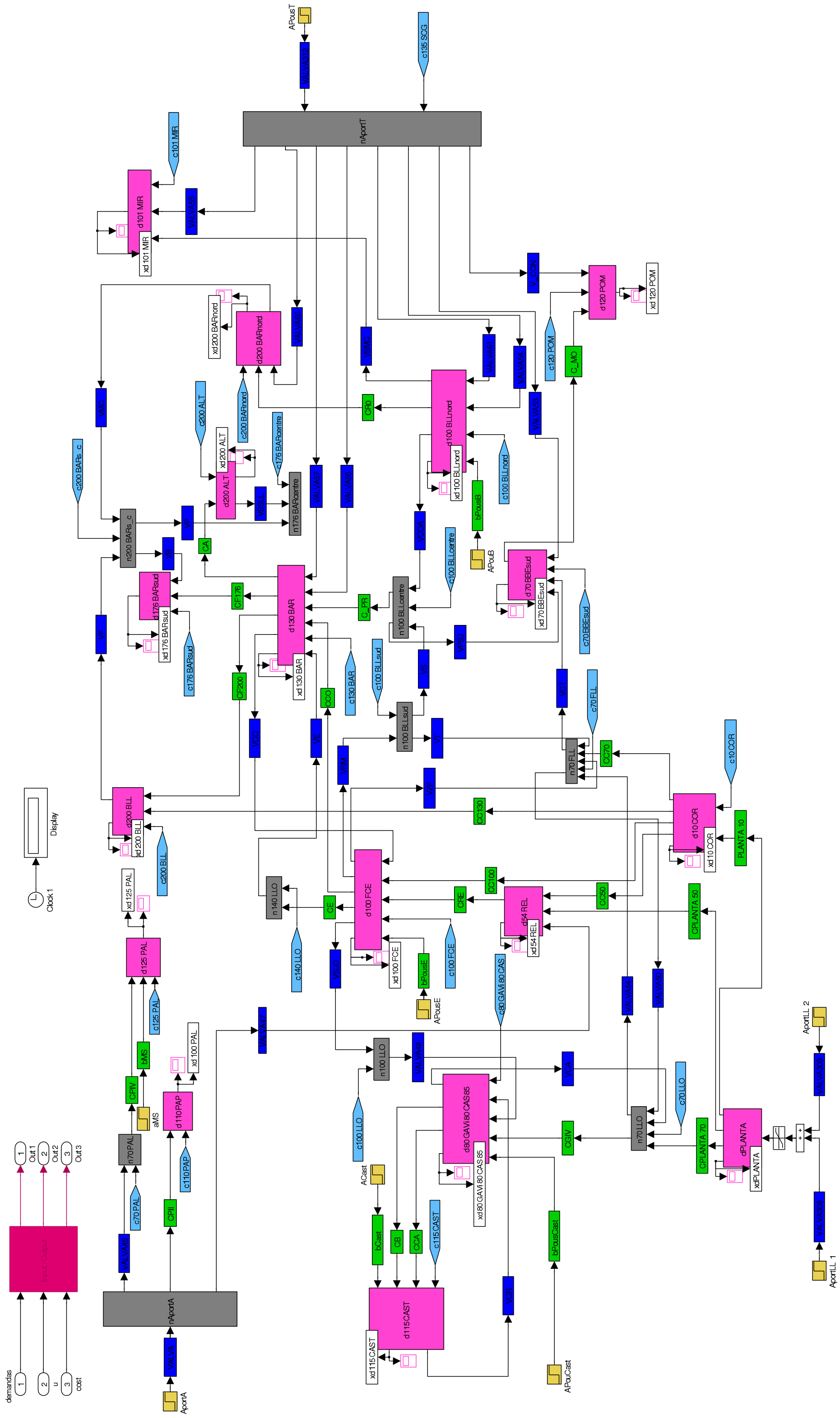


Figure 22: *Aggregated Barcelona drinking network model in* SIMULINK®

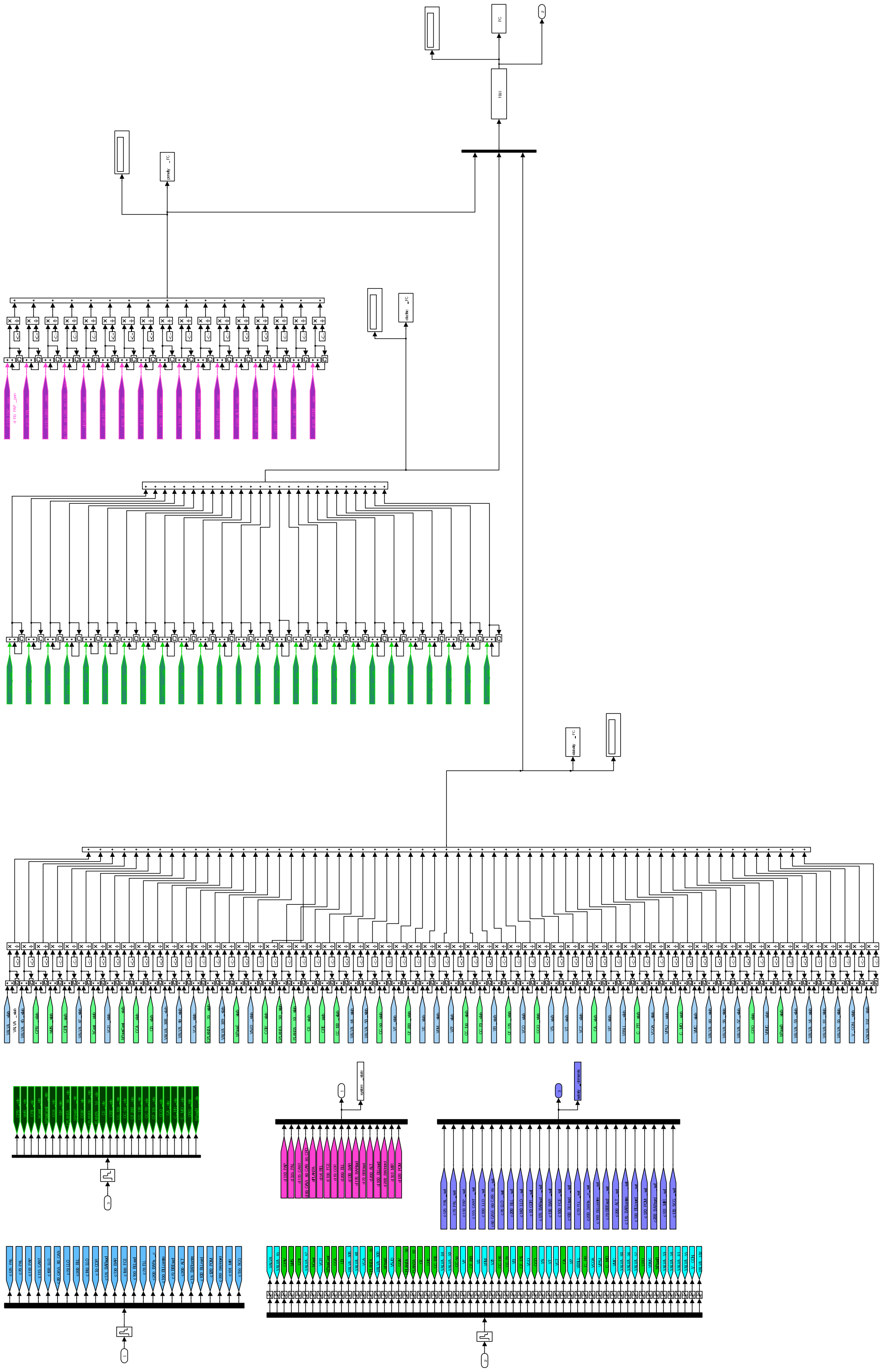


Figure 23: *The connections built into the block input/output*

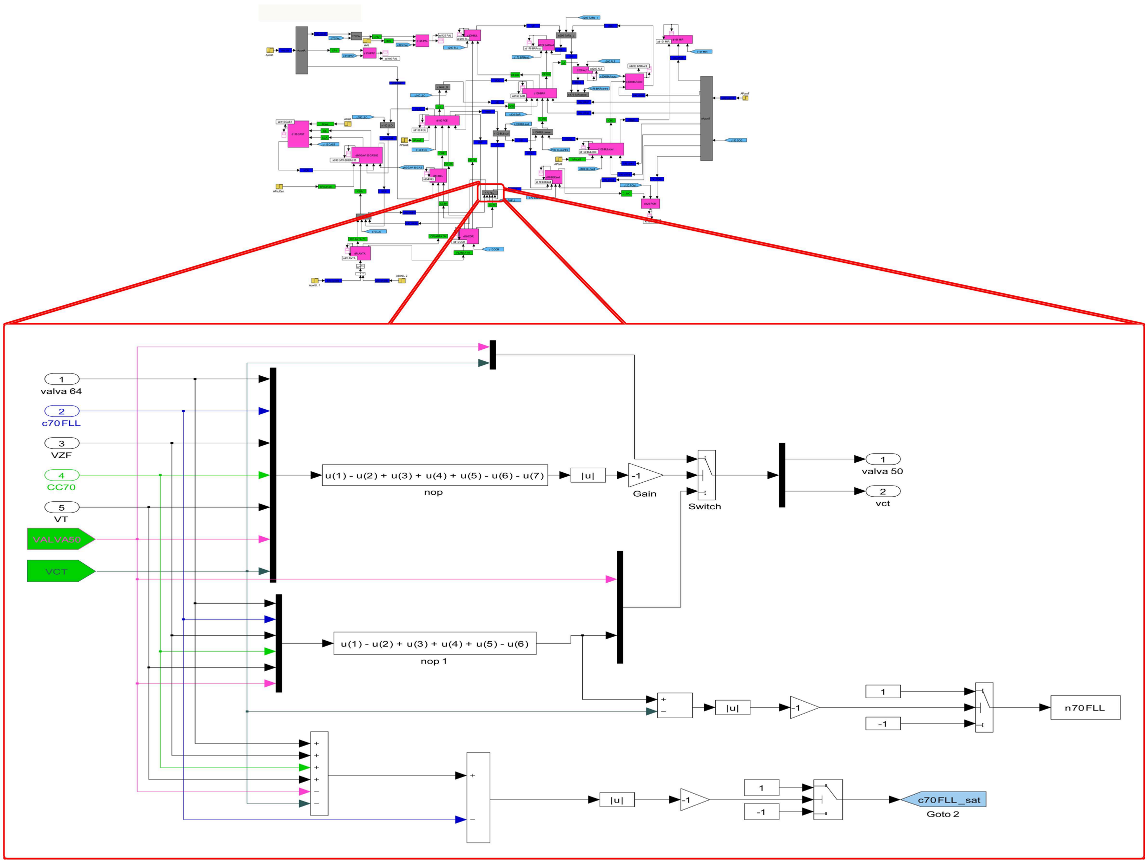


Figure 24: *Node implementation structure: in particular it is showed the implementation of the element n70*FLL

The tank d130BAR has 5 inputs and 5 outputs which interact in the computation of the volume level.
In this case, in the block equation it is found:

$$u(1) - u(2) + u(3) + u(4) + u(5) + u(6) - u(7) - u(8) - u(9) - u(10)$$

where the numbers into the bracket indicate the order of the multiplexer inputs. The second input is the demand sector c130BAR and it is the reason of the minus sign before the $u(2)$.
The tank equation, implemented for this element in the simulator, corresponds to the controller equation:

$$\begin{aligned} V(t+1) = V(t) + \Delta t\,(& u_{30}(t) + u_{38}(t) + u_{45}(t) + u_{51}(t) + u_{52}(t) + \\ & - u_{29}(t) - u_{36}(t) - u_{37}(t) - u_{42}(t) - d_{12}(t)) \end{aligned}$$

where it is used the name used by the controller.

6 Model Predictive Control of Barcelona Water Network

In this section, the model predictive control of the Barcelona water is presented. The model predictive controller uses a multi-objective cost function, which reflects the control strategy of the water network. The controller computes the optimal solution with a prediction horizon and demand forecast of 24 hours. t any time, only the first value is used and at the next time starts a new computation. The results are obtained interfacing the simulator described in the Chapter 5 with the PLIO tool and the MATLAB® controller developed in [7]. The last part shows the differences obtained using the MPC controller implemented in MATLAB in centralized and decentralized form.

6.1 MPC Cost Function

As we have already discussed, water networks are very complex multivariable systems. In order to improve their performance, predictive optimal control provides suitable techniques to compute optimal control strategies *ahead in time* for all the flow and pressure control elements of a water system, as discussed in Chapter 3. The optimal strategies are computed by optimizing a mathematical function describing the operative goals in a given time horizon and using a representative model of the network dynamics, as well as demand forecasts.
The cost function of a MPC controller, as it has already explained, is composed by several objectives. In this study, the cost function includes three objectives, which have to be minimized at the same time:

1. **Security term** FCS, that gives a penalization when the water level in the tanks goes below the security level. It is a operation-safety cost associated to not satisfying desired security storage volume in the tanks. The desired volume in the tank represents the security level which is needed to guarantee the demand satisfaction. Then, this criterion aims to the maintenance of appropriate water storage levels in the network for demand satisfaction. This level is chosen using the results obtained in the simulator, presented in

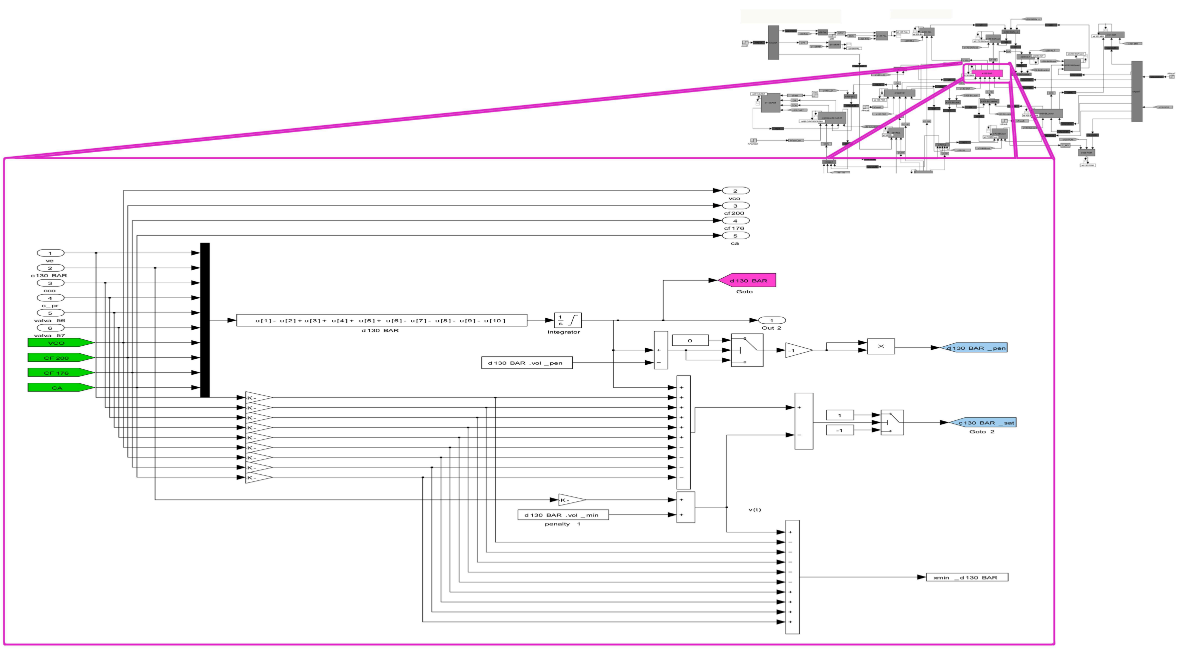


Figure 25: *The tank implementation structure: in particular it is showed the implementation of the element d130*BAR

the Chapter 5 where in the tank element is computed the minimal volume to satisfy the demands. Through this function, the objectives regarding the satisfaction of the demands and the preservation of a security level in the tanks, described in the Section 4.1, are satisfied. It means that with this water storage level, it is guaranteed that the demands could be satisfied. This function is obtained implementing the equations (14) and (17).

2. **Economical cost** FCP considers the economical cost associated to the supply, treatment and transport of the water.
   The water cost is usually related to the acquisition and treatment, which may have different price relied to different sources, and to the elevation, affected by electrical costs which may change during the day. This term assures that the cost should be as low as possible, allowing to satisfy the economical cost objective.

3. **Stability** FCE of the pumps and valves. This term penalizes the continuous set-point changes that could damage the actuators. This function is determined by the equation (15), where it is considered the changes in the actuator set-points in two consecutive time instants. It is important to keep in consideration, also, that the operation of water potabilisation plants usually requires very smooth flow set-point variations, so in this case, it could be necessary to use a bigger weight.

6.2 Evaluation of the Real Cost in the Water Network Simulation Environment

Using the simulator environment developed in Chapter 5, a network of Barcelona has been created. The simulator allows to compute the real value of the objectives explained above. The three terms of the cost function are computed in the input/output block where it is saved the total value of the cost function and the value of every objective (Figures 13 and 23). Every objective is computed in the way that the PLIO tool does. However, in the simulator we do not have implemented the normalization used in PLIO, in order to obtain the real value.
In the following a detailed description of the cost function implementation in the block input/output is presented.

Stability objective : this term includes all the actuators, and, how it was explained in the Chapter 5 in particular in the Section 5.2.2, every element contributes with a squared value. The squared stability term for every actuator is normalized dividing by the corresponding squared maximal flow. Then, it is summed for every sample time and every element to obtain the total stability term of the network. At this point, it is possible to consider this term directly, or after the division by the number of the elements (in this case the number of the actuators) and the length of the simulation horizon. In second case, it is found a term which could be compared with the other objective, as in PLIO tool, indeed it is independent of the number of elements. To compare the performance of the controller in different scenarios we use this last term.

Security objective : this term includes all the tanks, that is, the penalty term computed for every tank in the corresponding subnetwork. The security term is computed in two different ways since the simulator should reproduce the cost function implemented in PLIO tool, or in the MPC controller developed in MATLAB®. In the first case, the squared penalty term computed in the tank subnetwork is divided by the corresponding squared penalty volume. In the second case, the computation of the penalty term in the tank subnetwork is also changed. In fact, the implementation of a non-linear function in the controller it is not quite simple, and for the moment the controller (see the [7]) presents the typical set-point MPC cost function. The system state has to track a reference being penalized every time that the reference and the state are different. So, in this case, we obtain a penalization also when the level is over the security level. This squared term, moreover, it is divided by the squared of maximal volume of the corresponding tank. In practise, the cost function implemented in PLIO tool is preferred, but it is interesting also to study the difference obtained considering the typical MPC objective. As in the stability term, it is possible and useful to normalize with the number of tanks and the length of the simulation horizon. As in the stability term, the normalized value is used in the study of the controller performances in different scenarios.

Economical term : this term includes all the pumps and valves which follow a source. It is computed, simply, considering the total cost for every pump, or valve, in the simulation horizon. To find the total economical cost of the network, all individual costs associated to every element are added together such that a value in euros is obtained. In this case, is non sense to normalize it by the number of elements because the interest is in the total cost. To study in detail the behaviour of the network, this term is divided in two different subcosts: one which considers the water cost and the other the electrical pumping cost.

All these terms represent the indicators to tune the MPC controller. In the following study the differences, in term of cost function, obtained changing the parameters of the network, like, for example, initial condition, penalty level and weights in the PLIO simulations are analyzed. The results obtained with the MATLAB® MPC controller are reported in Section 6.5.

6.3 Implementation of the Cost Function in PLIO

The multi-objective cost function in PLIO is thus chosen as the addition of a weighted sum of the terms, each one representing an objective:

$$FC = \omega_1 \cdot FCS + \omega_2 \cdot FCP + \omega_3 \cdot FCE$$

where:

- ω_1, ω_2 and ω_3 are the weights, which correspond to the priority level.
- FCS is the security cost;
- FCP is the economical cost;
- FCE is the stability cost.

The weights are selected on the basis of the priority order. It is possible to make some tests to evaluate which is the best choice of priority level. A technique that could be used to search the optimal value of each optimization term is considering one objective at a time. Unfortunately, in this model it is not possible to use this technique, because, considering only one objective at time the computation time increases a lot. This happens because the cost function in this way becomes very flat and so it is very difficult to find the optimal solution. The optimizer, in the majority of cases, chooses a local solution, and not a global optimal solution.
In the following, a detailed description of the objective function used in PLIO tool is presented. These equations use the variables created by PLIO tool, which have already described in the Chapter 4.
The results computed in this way represent the value obtained for every objective function at the end of the predictive horizon. In fact PLIO tool calculates the value of the cost function considering the 24 hours of the prediction horizon. However, currently, this tool does not provide directly a real value for the cost function but only an estimated in 24 hours. For this reason, it is also important the computation done using the SIMULINK® model.

6.3.1 Security Term (FCS)

The security term assures that the water level into the tanks remains above a selected volume, called penalty value. This function considers the level of every tank into the model. Every tank may be weighted in this function with a different coefficient. This weight is the value that appears in the tank property's windows in PLIO tool, in Editor mode in the Section 2.2.1.
As it has been done for every element in the Chapter 2 in the Section 2.2.2, now it is shown how the PLIO tool, with the aid of GAMS solver, computes this function and what types of normalization are used.
The security function is defined like a weighted sum of variables **Vbajo**xx, computed in the equation (7), that indicates, for each tankxx (with $xx = 1, \cdots, 17$), whether the security level is satisfied or not:

$$FCS = \frac{\sum_{i=1}^{17} \omega_i \mathbf{Vbajo}_i}{17} \tag{27}$$

where ω_i is the weight concerning the tank i.
Moreover, the equation (27) considers the normalization dividing by the number of elements. In fact, there is a division by 17, where 17 is the number of the tanks in the model. It is important to remember that the term **Vbajo**xx includes, for every tank, the sum along all the prediction horizon and the respectively normalization.

6.3.2 Price Term (FCP)

The FCP term represents the water and electrical costs, that are usually related to acquisition and treatment, which have different prices at different sources due to the different chemical treatments and electrical pumping. The pump electrical cost, necessary to drive the water at a bigger elevation, has a different value at different hours of the day, as it is possible to see in the Tables 3, 4, 5, 6, 7. Analogously to the security function, every pump or source may be weighted by a different coefficient, inserted in the property window of each element.
The price function is a weighted sum of variable concerning the sources and the pumps:

$$FCP = \frac{\sum_{i=1}^{9} \omega_i \mathbf{sum}_i}{9 \cdot 24 \cdot \max\_unitary} + \frac{\sum_{j=1}^{26} \omega_j \mathbf{sum}_j}{24 \cdot 26 \cdot \max\_elect} \tag{28}$$

where the variables $\mathbf{sum}_i$ and $\mathbf{sum}_j$ are respectively, obtained in equations (8) and (11), while ω_i and ω_j are the weight concerning, respectively, the source i and the pump j. The variables name $\max\_unitary$ and $\max\_elect$, respectively, correspond to the maximal unitary cost which appears considering all the sources, and the maximal electrical cost imposed by all 26 pumps. The number in the denominator is used to normalized the objective. In fact, every weighted sum is divided by the number of the elements and the prediction horizon length. All these normalizations allow to obtain values between zero and one. In this way, it is possible to compare the different objectives and using different weights reflect the priority order.

6.3.3 Stability Term (FCE)

The stability term preserves the actuator from both continuous and abrupt variations. This function is a weighted sum of variable concerning the actuators which should be stabilized. All the actuators, pumps and valves, have to be stabilized. To reach the goal of making the valves stabilizable, the motorized valves, which are elements of the PLIO software tool, are used.
The mathematical form of this function, considering the terms generated by every element, is the following:

$$FCE = \frac{\displaystyle\sum_{i=1}^{p} \omega_i \cdot \mathbf{Est}_i + \sum_{j=1}^{v} \omega_j \cdot \mathbf{Est}_j}{p+v} \tag{29}$$

where $\mathbf{Est}_i$ and $\mathbf{Est}_j$ are the variables obtained, respectively, from the equation (9) and (10) for every pump or valve; p and v are the numbers of the pumps and the valves which have to be stabilized, respectively. The variables $\mathbf{Est}_i$ and $\mathbf{Est}_j$ have already been normalized with respect to the prediction horizon.

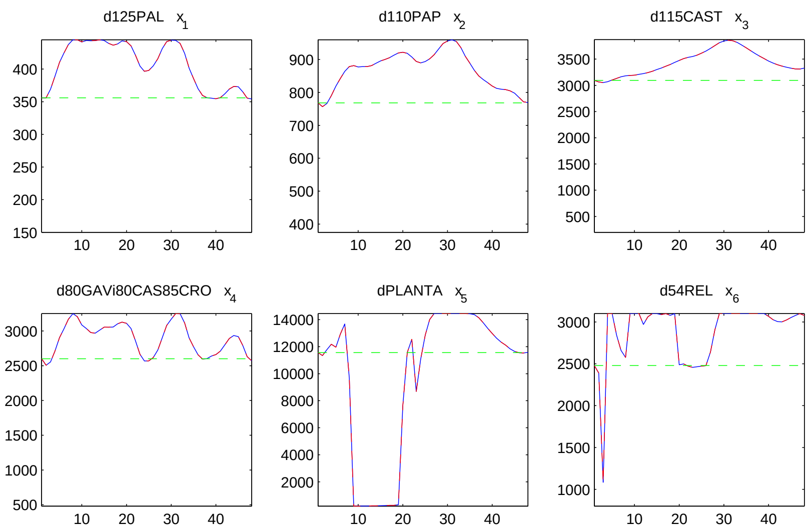
6.4 MPC Results obtained using PLIO

This section deals with the simulation results obtained using the PLIO tool. Several test scenarios have been chosen to show the potential of predictive optimal control tool for computing control strategies in complex operational situations. In addition, the results of these different scenarios allow to select the element and cost function weight and the other parameters, as the penalty or initial volume, in order to obtain a network with as possible.
Each scenario shows the simulation of 2-days (48 hours), and in particular they include data relative to 2 days of hourly demands. These demands represent the request in a standard daily situation. Indeed the hourly values in one day are obtained considering the monthly amount of water request. The hourly values reflect, as is possible to view in the Figure 9, the same daily forecast of every demand sector.
The two days of demand considered have the same profile, but in the future study could be interesting to use a different daily profiles for every different day in the week. This will allow to reflect the different behaviour of the users in the different day, as, for example, a Monday (a working day) or a Sunday (a holiday day). It would also be interesting to study the user behaviours in presence of some public events, in order to manage the forecast demands, in several situations.
The first scenario shown is used as a starting point in the tuning process and it was used to find the way to improve the control results following AGBAR suggestions.

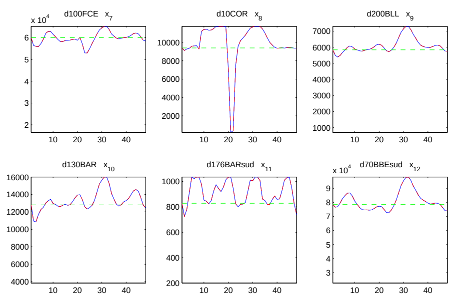
6.4.1 Scenarios as Starting Points

The scenario I represents the starting point for our analysis. In fact, at the beginning, of the study it has been considered a typical scenario which allows to analyze the network behaviour and, from these results, decide the better optimization strategy and parameters.
This scenario is planned using the 80% of the maximal volume of every tank both as its initial volume and penalty level. The results obtained are used to find a penalty volume value less restrictive which allows to save more water and, then, money. This penalty value, as it has been indicated before, is computed thanks to the simulator implemented in the SIMULINK® environment.

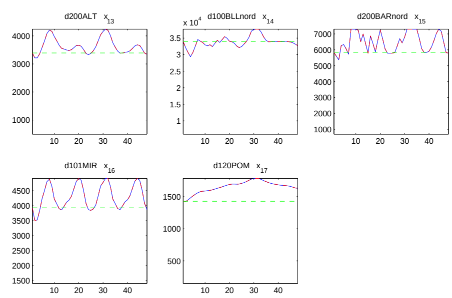
The weights are also selected in a very simple way in order to consider all objectives with the same priority both the security and the stability term (using a weight of one), while the economical cost (using a weight of zero) was not initially considered. The resulting cost function has a very flat form, which increases a lot the computation time (for a simulation of 48 iteration it was necessary about three hours and half). In fact, it is quite difficult to find a global minimum, and, the optimizer stacks in a local minimum encountering some numerical problems, which were indicated, in the files generated by PLIO, as "*non-optimal* ". The optimization, if does not reach the optimal solution, is stopped by the GAMS solver after an imposed interval of time, which is 5 minutes by default. But to try to obtain a minor number of non optimal values it was increased to 15 minutes.



(a) *Volume evolution of the first six tanks (from x_1 to x_6)*



(b) *Volume evolution of six tanks (from x_7 to x_{12})*



(c) *Volume evolution of five tanks (from x_{13} to x_{17})*

Figure 26: *The volume evolution of a tanks in the scenario I: in blue is showed the values computed by the simulator, in the broken red line that calculated by PLIO tool, and the broken green line indicate the 80% of the maximal volume, the penalty value.*

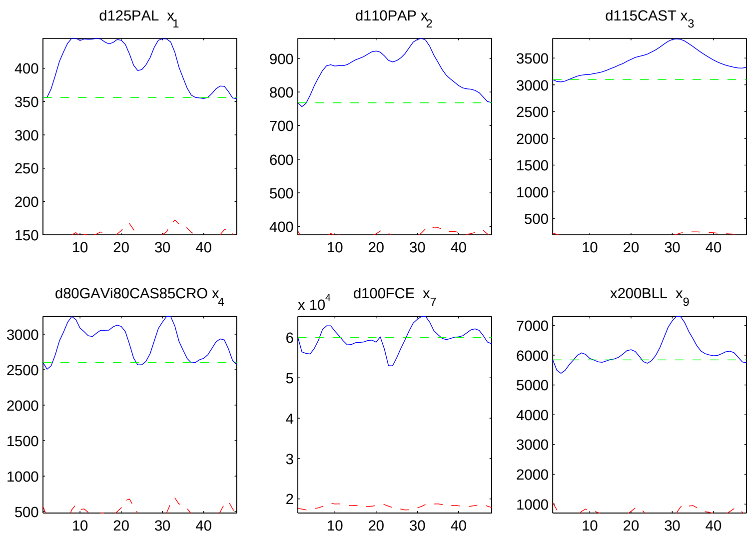
The evolution of the 17 tank volumes, obtained in this case, is presented in the Figures 26 where it is possible to compare the evolution computed by PLIO (in the broken red line) and by the simulator developed in Chapter 5 are completely overlapped for every tank. In the broken green line the 80% of the maximal tank volume is indicated, which, in this scenario, represents both the initial volume and the penalty level. The range of the y-axis in all the plots reflects the

physical range of the tanks volume: starting at the minimal volume allowed until the maximal, as it is reported in the Table 1.
The SIMULINK® model, besides to simulates the evolution of the tank volumes in the simulation horizon, computes the value for every objective of the cost function. In these scenario, these values do not have much sense, but they are a useful starting point to compare the various scenarios. In this case since the economical cost is not included in the objective function it has a big value. The values obtained for every objective are:

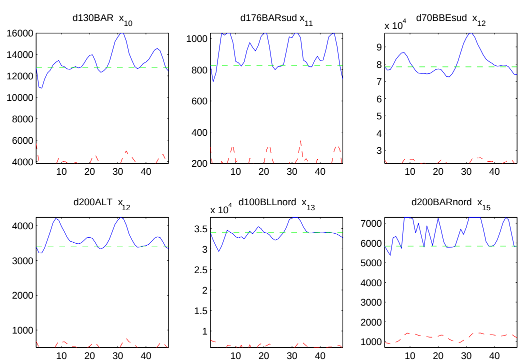
$$\begin{aligned} FCE &= \frac{10,25}{61 \cdot 48} = 0,0035; \\ FCS &= \frac{13,368}{17 \cdot 48} = 0,0164; \\ FCP &= \frac{174450}{2} = 87.225 \text{ euros} \end{aligned} \tag{30}$$

The first two objectives, the stability and the security term, are simply a number, in fact they are obtained normalizing the total resultant after a 2 days simulation for the number of the elements and for the length of the simulation horizon. The economical cost, otherwise, is expressed in euros, and the value after the equal sign represents the cost in one day. So, with this chosen of weights the total expense every day is about 87.000 euros. This value is obtained summing the water and the electrical costs. The cost is very influenced by the initial condition, and for this reason in the Table 14, the electrical and water cost are reported separately for every day. Logically, the first day is more affected than the second by the water stored in the reservoirs at the beginning of the scenario. It is also very interesting to study the different behaviour of these two cost components using different strategies.
The optimizer minimizes the multi-objectives cost function, which it is obtained summing every objective multiplied by its weight: in this case it is, simply, the sum of the two terms FCE and FCS, since the weights are one both. This scenario was useful, mainly, to determine for each tank the minimal volume necessary to satisfy the demands at every time. This computation was implemented in the SIMULINK® simulator. Figure 27 shows, in the broken red lines, the evolution of these minimal volumes. These evolution are computed considering the demands and the actuator set-points at every instant, so they depend on the particular scenario. However, the levels obtained are useful to determine a minimal penalty volume. The security level imposed at the 80% of the maximal is too restrictive, according to these results, since a lot of water is stored without any real reason and it is a waste of water and energy. The minimal volume could only be computed for the tanks which directly supply water to a demand sector.
The other tanks (more precisely: dPLANTA [x_5], d54REL [x_6] and d10COR [x_8],) are used like a buffer to store water generated by the production plants and to supply the other parts of the network connected to them.
To compute a penalty value for every tank based on these values, that allow to have a feasible solution, the pick of the volume indicated with the broken red line in Figures 27 has been considered, by increasing it a 20%, as it is shown in the Table 12. This sum allows to obtain a value with a degree of security that could be valid also in other scenario.
The graphical results of these computations are shown in Figure 28, where it is possible to observe that the broken line red is always under the big magenta line which indicates the penalty volume.
These values are a lot underneath the 80% of the maximum volume, indeed all are under the 50%. So, it is reasonable thinking to obtain, using these levels, a gain in terms of economical costs. Moreover, the security became less restrictive, and so it is more easy to respect this constraint and the value of this objective could decreases.

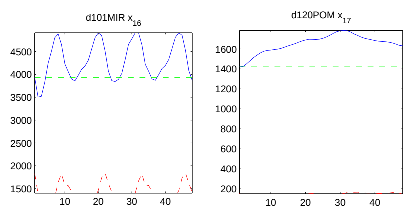
The sources in the network have different prices: in particular the underground sources have a bigger cost compare to the superficial ones. Without including the economical cost, the optimizer is free to chose where the water is taken from, without any rule of decision. As it is possible to observe in the Figure 29, in this scenario also the underground sources are used, which, logically, causes an increase of the cost. In the Figure 29, the first three plots show the value of the three superficial sources, while the other are three examples of the underground sources and their use.
Scenario *II* differs from the scenario *I* only in the use of the penalty levels presented in the Table 12, and also in the in modification of the stability weight in the cost function. For the three tanks that do not directly supply a demands sector, the maximal value has been chosen as penalty level. In this scenario the initial volumes continue to be set at the 80% of the maximal volume of every tank.
The objective function still does not include the economical term and although changing the objective weights, the cost function form becomes a bit more restrictively, the computational time is still big and presents a lot of non optimal values, but decreases a bit with respect to the



(a) *Minimal volume trend of six tanks*



(b) *Minimal volume trend of six tanks*



(c) *Minimal volume trend of two tanks*

Figure 27: *The minimal volume requested to satisfy the demand it is displayed in the broken red line, while the blue line indicates the volume trend in the scenario I*

Tanks name	Vector state	Minimal volume pick	Penalty volume
d125PAL	x_1	172,40	200
d110PAP	x_2	399,83	500
d115CAST	x_3	256,01	400
d80GAVi80CAS85CRO	x_4	693,10	850
d100FCE	x_7	18.917	25.000
d200BLL	x_9	1.005,3	1.500
d130BAR	x_{10}	5.449,3	6.500
d176BARsud	x_{11}	3.332,66	400
d70BBEsud	x_{12}	26.787	30.000
d200ALT	x_{13}	755,82	1.000
d100BLLnord	x_{14}	8113,5	10.000
d200BARnord	x_{15}	1.404,7	2.000
d101MIR	x_{16}	2.070,5	2.500
d120POM	x_{17}	160,54	400

Table 12: *Tanks penalty volume level computed summing to the pick its 20%. All volumes are expressed in* $[m^3]$

scenario *I*. For these reasons, the results obtained in this scenario are useful for a comparison with the other scenarios presented in the Table 14 but not very interesting to be studied alone. The values for the cost function reported in the Table 14, could be surprising, because the cost resulting is bigger although the penalty level is lower. This is due to the fact that, no considering the economical objective, the selection of which source or which pump to use is complete unconstrained. Thus, the obtained feasible solution is not the optimal solution.

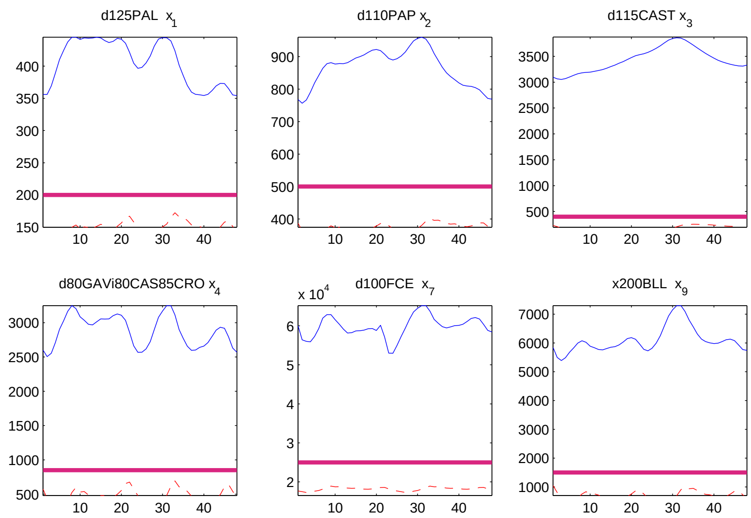
6.4.2 Initial Condition

An important issue in the improvement of the predictive control results is to understand the effect of the initial condition in the tanks in the cost function values. With this aim, we have simulated three scenarios (called *III*, *IV*, *V*) which have all the same weight and security level in the tank, but with different initial conditions each one.
The weights used in the cost function are: 10 for the security term, 0.1 for the stability and 100 for the economical objective. This could be considered a good choice, since in the solution provided by the optimizer does not appears any “on-optimal” value. Then, in every of these three scenarios, the optimal solution is obtained improving also the computation time. In fact, in less than one hour and half for every scenario (with 48 iterations) is required. There are some small difference depending on the initial condition, which makes more or less restrictive (flat) the cost function.
The penalty levels used in each one of these scenarios are those presented in the Table 12, in order to not storing a lot unnecessary water in the tank.
The initial conditions are chosen in this way:

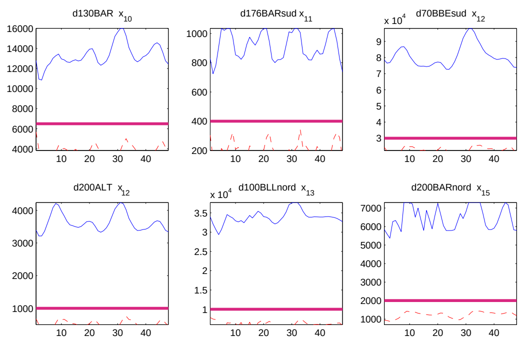
- in the scenario *III*, the 80% of the maximal value in every tank is imposed.
- in the scenario *IV*, the volume obtained at the end of the scenario *III* is imposed, in order to try to obtain the same evolution in every day. In order to guarantee perfectly such repetitive behaviour, it would be necessary to use a bigger number of iterations, to determine the right initial condition, but also in this case we have obtained some interesting results.

- in the scenario V, the penalty level used.

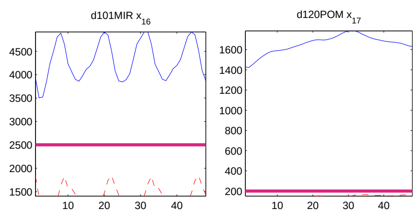
The different evolutions of the tanks with the different initial condition are shown in the Figure 30 and 31, where, respectively the tanks evolution in the scenario *III* and V scenario are presented. According to what it is explained before, the effects of the initial condition are evident in the daily water cost and particularly in the economical term. The water necessary to satisfy the demand in the first day of simulation change according to the initial condition: when the system starts with the tank at the 80% less water is needed to satisfy the demands (scenario *III*) which implies a save of money. Decreasing the initial volume, logically, increases the cost, as in the scenario *IV* and V. The numerical values are reported in Table 13. In particular in Table 13, it is shown the value of the three main objectives and, moreover, the difference between the first and the second day in the water and electrical cost.
In order to complete the analysis of these three scenarios, it is useful to see how they use the sources. Now the economical cost is included in the cost function so, in a standard day as the one to that is considered, the optimizer uses only the superficial sources. This behaviour is



(a) *Penalty volume chosen in six tanks*



(b) *Penalty volume chosen in six tanks*



(c) *Penalty volume chosen in two tanks*

Figure 28: *Penalty volume chosen considering the minimal volume showed in the* Figure 27: *in the broken red line it is found the minimal volume needed to satisfy the demand, in the blue line the volume evolution in the scenario I and in the magenta line the penalty volume computed.*

Scenario	Initial condition	Objectives values FCS	FCE	FCP	Water cost 1 day	2 day	Electric 1 day	2 day
III	80%	$1,07 \cdot 10^{-8}$	0,0024	126.620	49.205	60.531	8.332	8.546
IV	rep	$1,02 \cdot 10^{-6}$	0,0027	133.550	55.379	61.123	8.496	8.554
V	pen	$8,97 \cdot 10^{-6}$	0,0032	142.880	63.832	61.979	8.546	8.520

Table 13: *Cost value obtained varying the initial condition: the scenarios III IV and V*

confirmed by Figure 32, where the use of all superficial sources and some underground sources is reported, in the scenarios *III* and *V*.
In fact, the plots in the second row, which correspond to case of the underground sources, are always zero. This assures a save of money, and, moreover, that the network behaviour is close to the optimal one.

6.4.3 Penalty Level

Another parameter which could modify the cost function values is the penalty level. In order to obtain more reliable results than the case of the first two scenarios, where the big number of *non-optimal* does not allow a significantly study, we have implemented a new scenarios (the *IX* in the Table 14).
The scenario *IX* has the same initial condition and weights than scenario *V* but change the penalty volume to the 80%. This change is reflected in all the three objective that increase. The major difference, logically, appears in the security level. In fact, this is the first scenario where the initial condition in under the penalty level. Then, at the beginning, we have a penalization for every tank, and a penalty level of the 80% is a very strong restriction. This objective also increases the economical cost. In fact, the water is used not only to satisfy the demands, but also to fill in the tanks until the 80%. To do this, it is also necessary to use more the pumps and the valves and so it is for this reason that the stability term increases. As for the other scenarios, the values obtained are reported in Table 14. The economical variation is mainly due to the water cost in the first day. These results confirms our idea, explained before, that the water is used also to fill the tank. Since the second day the economical costs are comparable to

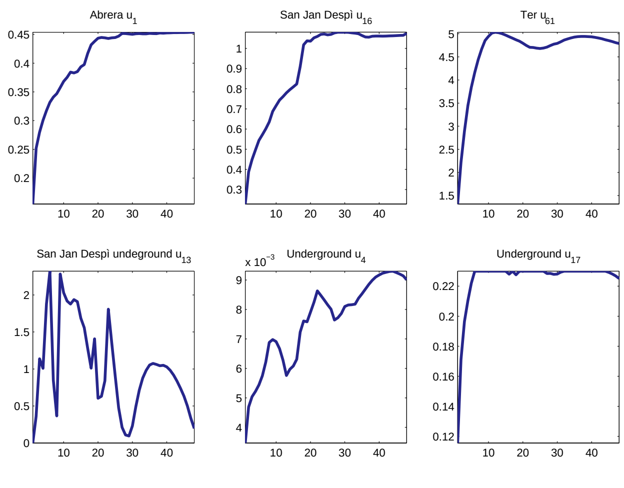
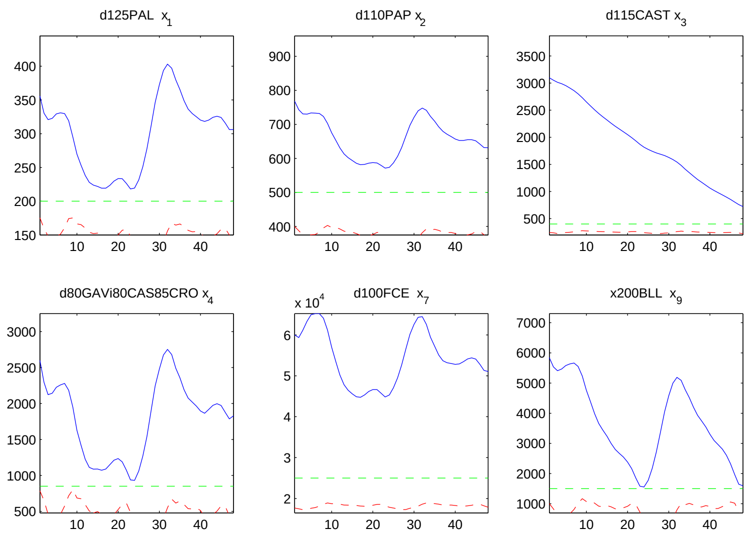


Figure 29: *The use of some source in the scenario I*

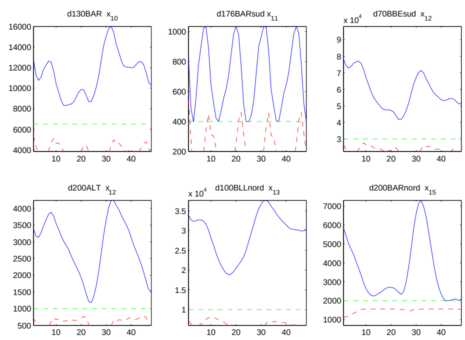
the costs obtained in the scenario V.
The Figure 33 shows the evolution of the tanks obtained in scenario IX. The convention used is the same for the others figures, as for example in Figures 30 and 31. The different penalty level influences a lot the evolution of the tanks, altouugh in this scenario, at the end of the two days, all volumes are all over the security level. In this way, we store more water which it is reflected in the cost of the water that increases from the scenario V to the scenario IX.

6.4.4 Priority and Weights

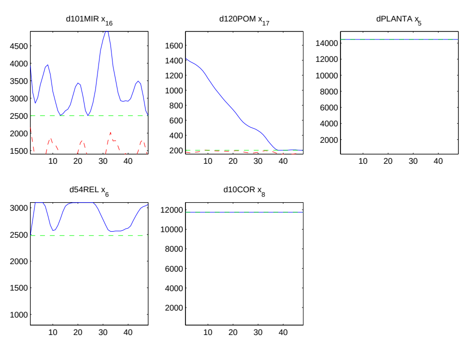
Finally to close the study of the MPC using PLIO, the analysis regarding the change of the priority order in the objectives is presented in order to discover the best wight choice to obtain in the minor time the best solution for all three objective. In fact, we search the best compromise between the objectives, taking into account that, in reality the main interest is to minimize the



(a) *Evolution volume in the six tanks*



(b) *Evolution volume in six tanks*



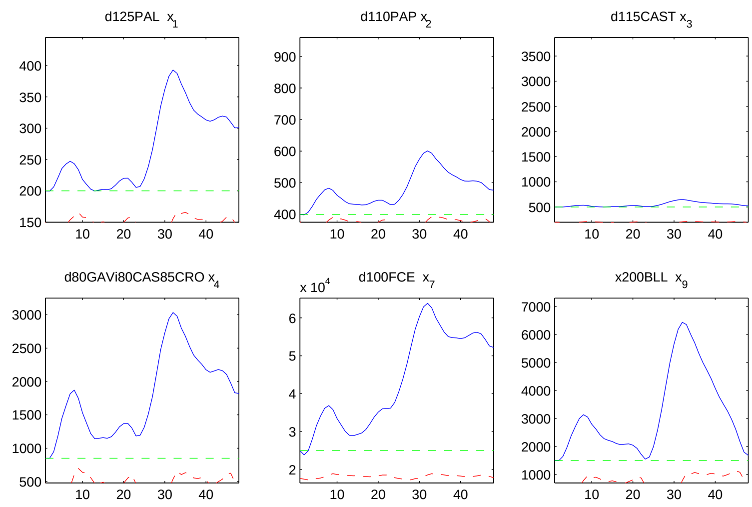
(c) *Evolution volume in five tanks*

Figure 30: *Volume evolution obtained in the scenario III: the minimal volume needed to satisfy the demand is, as usually, in the broken red line, while in the blue line is shown the volume evolution in the scenario III and in the broken green line the penalty volume. The last three tanks have not the minimal volume, because they are considered like a buffer.*

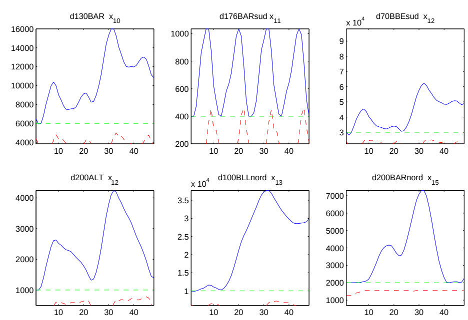
economical cost. The user is not interested in how much water is inside the tanks but how much expensive is the supply of the water in order to satisfy the demands. Moreover, for the user, it is important to avoid the damage of the actuators, due to a incorrect use, which could cause disruption of the supply and unforeseen expensese.
In this study, we have considered scenarios with the same initial condition and penalty level, while changing the weights and so the priority order. Observing the Table 14, this analysis has been done between these scenarios:

1. *IV* and *VII*;
2. *V*, *X*, *XI* and *XII*.
3. *III* and *VIII*;

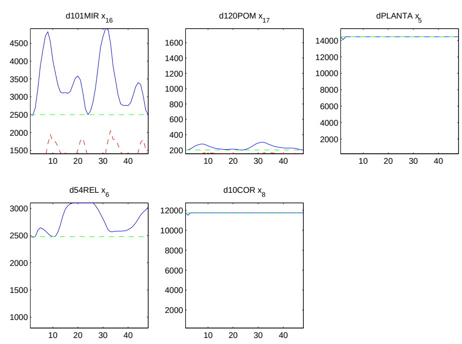
The first scenario of every group (the *III*, *IV* and *V*) is always a scenario described in the Section 6.4.2, where the weights used give more priority to the economical term, followed by



(a) *Volume evolution in the six tanks*

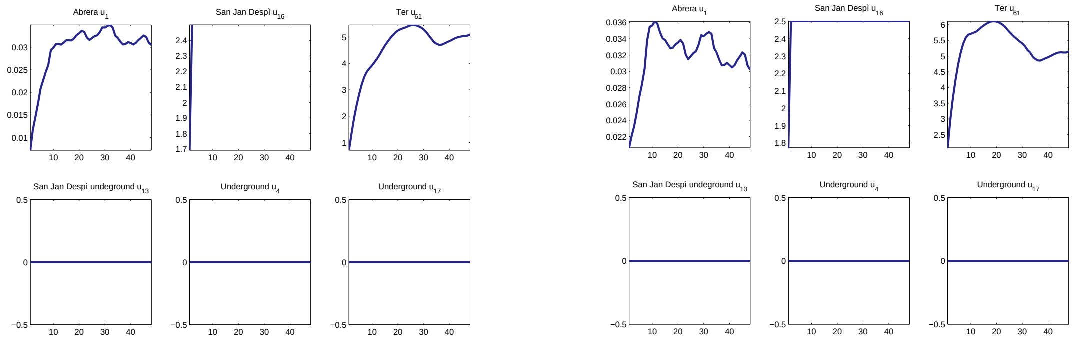


(b) *Volume evolution in six tanks*



(c) *Volume evolution in five tanks*

Figure 31: *Volume evolution obtained in the scenario V: the minimal volume needed to satisfy the demand is, as usually, in the broken red line, while in the blue line is shown the volume trend in the scenario V and in the broken green line the penalty volume. The last three tanks have not the minimal volume, because they are considered like a buffer.*



(a) *The use of some source in the scenario III*

(b) *The use of some source in the scenario V*

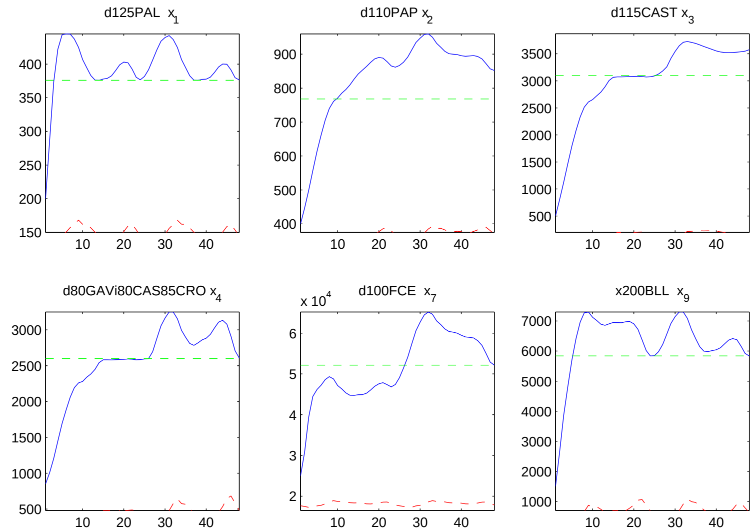
Figure 32: *The use of some source in the scenarios III and V*

security and, at the end, by the stability.
In the first group, the priority order has been maintained and it has been changed only the weight values. The values obtained are very similar what allows to state that without changing the priority order the values of the weights are not very important.
The real interesting study is represented by the other two comparisons where changing the weights is also changed the priority order. As it is reported in the Table 14 at the beginning in the scenario *X*, we have exchanged the weights between the penalty and the stability term maintaining the economical cost as the highest priority.
Notice from Figure 34 that weighting less the security term, the MPC strategy is more risky since the tank evolution goes more often under the penalty level. This behaviour could generate in some particular scenarios some infeasibility. According to these plots, the value obtained computing the security level, which represent the risk in the tank volume, is the biggest between all scenarios presented in Table 14. The economical cost, instead, decreases as well as the stability term. In particular, the cost of the water decreases, since in this case, rarely, useless water is stored in the tanks.
In the comparison between the scenario *V* and the scenarios *XI* and *XII*, as well as between the scenarios *III* and *VIII*, we have tried to consider the economical cost as the less important term in the cost function. The scenarios *XI* and *VIII* use the same combination of the weights with a different initial condition, while the scenario *XII* use weights scaled in a different way. In these three scenarios, we have obtained an incredible increment of the economical cost, in all its components, without obtaining a significant improvement in the others terms. Therefore, we could state that the economical cost must have the most important role in the cost function. In this case, it is not very evident, also, the dependence on the initial conditions in the water cost.
Table 14 (the names are in the column Scen) reports, for every scenarios, the values of the all term in the cost function and weights, highlighting, moreover, the difference cost obtained in the two different days. The scenario *VI*, that has not been already explained, has been implemented to try to emphasise the economical cost, and considering with a very small weight the other term. The value obtained is not very relevant in terms of improvement in the economical cost. On the contrary, a value bigger is obtained compared to the one obtained in the scenario *IV*. The initial volume, called *rep**, it has been obtained after a simulation long 72 iteration with the same weights respect the scenarios *VI*. This choice, in these particular scenarios, is more repetitive than that obtained in the case *IV*. Moreover, the other terms are increased a bit respect with the other scenarios.

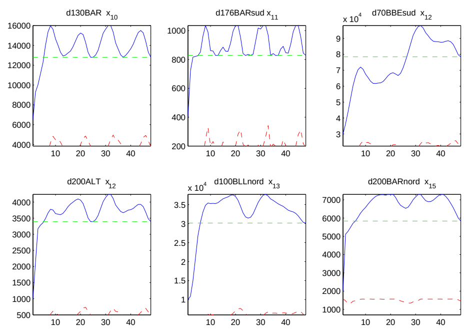
6.5 Results obtained by MPC Controller

In this section, another possible interface of the simulator developed in the Chapter 5 is presented. It shows that the simulator is a versatile tool, that it is able to obtain relevant results using whatever controller.

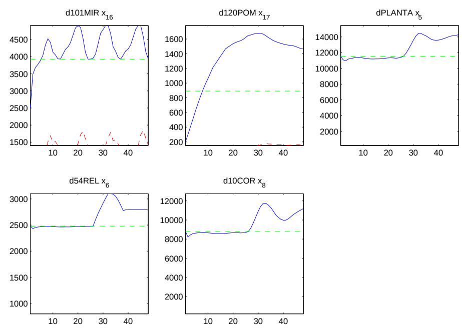
In particular, this section deals with the results obtained, in term of cost function, using the centralized and decentralized MPC controller developed in the MATLAB® described in [7]. In the report [7], an implementation of the centralized MPC for the Barcelona water network is presented, and, moreover, an implementation of a decentralized MPC control which allows to obtain good results, diminishing a lot the computation time, applying the control to three subsystem of the network. In the last case, the computational time is almost reduced to the half with respect to that one needed by the centralized controller. These two controllers have the same cost function, which, as is explained in the Section 6.2, is computed in a different way respect in the PLIO tool, regarding the security term. Moreover, the controller, by now, does



(a) *Volume evolution in six tanks*

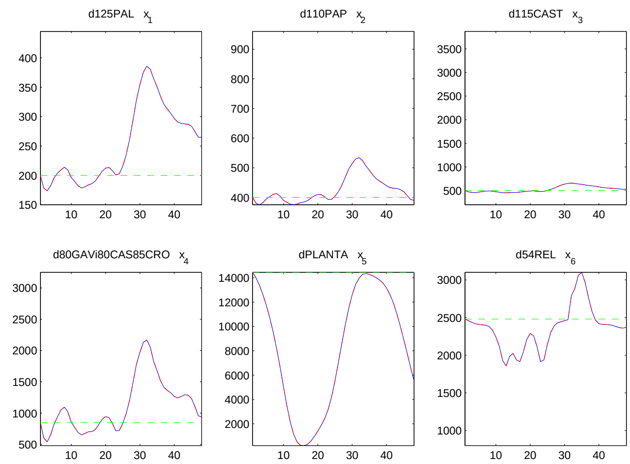


(b) *Volume evolution in six tanks*

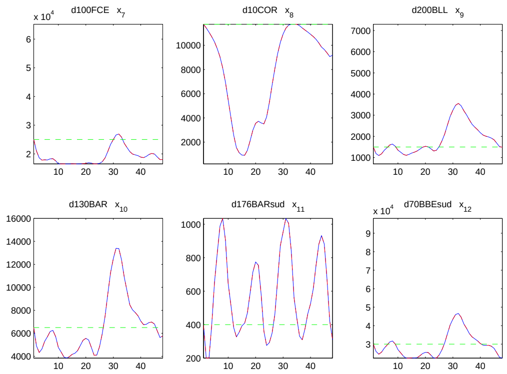


(c) *Volume evolution in five tanks*

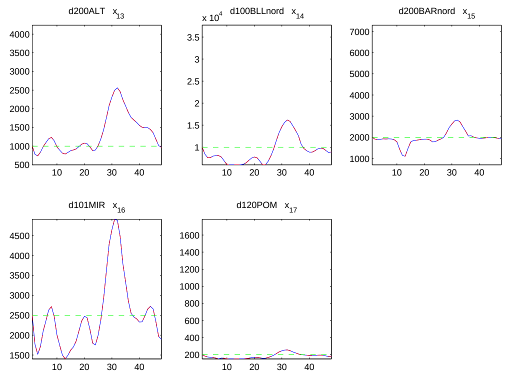
Figure 33: *Volume evolution obtained in the scenario IX: the minimal volume needed to satisfy the demand is, as usually, in the broken red line, while in the blue line is shown the volume evolution in the scenario IX and in the broken green line the penalty volume. The last three tanks have not the minimal volume, because they are considered like a buffer.*



(a) *Volume evolution in the first six tanks (from x_1 to x_6)*



(b) *Volume evolution in six tanks (from x_7 to x_{12})*



(c) *Trend volume in five tanks (from x_{13} to x_{17})*

Figure 34: *Volume evolution obtained in the scenario X: in the blue-green line is shown the volume trend in the scenario X and in the broken red line the penalty volume.*

Scen	**Pen**	**Init**	**Objectives weights**			**Objectives values**			**Water cost**		**Electric**	
	vol	**vol**	FCS	FCE	FCP	FCS	FCE	FCP	1 day	2 day	1 day	2 day
I	80%	80%	1	1	0	0,0164	0,0035	174.460	83.789	75.420	7.926	7.323
II	+20%	80%	1	0,1	0	$3,01 \cdot 10^{-4}$	0,0024	179.450	68.898	75.420	7.926	7.323
III	+20%	80%	10	0,1	100	$1,07 \cdot 10^{-8}$	0,0024	126.620	49.205	60.531	8.332	8.546
IV	+20%	rep	10	0,1	100	$1,02 \cdot 10^{-6}$	0,0027	133.550	55.379	61.123	8.496	8.554
V	+20%	+20%	10	0,1	100	$8,97 \cdot 10^{-6}$	0,0032	142.880	63.832	61.979	8.546	8.520
VI	+20%	rep*	1	0,01	10^7	0,0180	0,0122	135.100	57.845	62.093	8.086	8.079
VII	+20%	rep	10	100	10^4	$8,27 \cdot 10^{-4}$	0,0021	127.370	50.733	59.491	8.423	8.724
VIII	+20%	80%	1.000	100	0.1	$8,26 \cdot 10^{-5}$	0,0027	251.890	108.350	117.840	12.704	13.091
IX	80%	+20%	10	0.1	100	0,0160	0,0122	152.370	74.776	59.842	9.263	8.490
X	+20%	+20%	0.1	10	100	0,0488	0,021	134.140	57.504	59.249	8.003	9.386
XI	+20%	+20%	1.000	100	0.1	$8,87 \cdot 10^{-5}$	0,0027	251.980	108.350	117.840	12.704	13.091
XII	+20%	+20%	100	10	0.1	$8,91 \cdot 10^{-5}$	0,0026	253.260	111.320	116.350	12.849	12.738

Table 14: *Summary of cost function values in different scenarios with different weights obtained with PLIO tool.*

Sim	Pen vol	Cen/ Dec	Weights		Objectives values			Water cost		Electric	
			FCS	FCE	FCS	FCE	FCP	1 day	2 day	1 day	2 day
I	80%	cent	1	1	9,4981 *(0,0116)*	48,5806 *(0,0166)*	220.080	140.030	90.531	13.108	12.411
II	80%	cent	1	0,1	9,0197 *(0,0111)*	48,5207 *(0,0166)*	219.730	104.060	90.237	13.071	12.362
III	γ	cent	1	1	47,0024 *(0,0576)*	27,002 *(0,0092)*	197.850	85.466	89.923	11.975	12.487
IV	γ	cent	1	0,1	47,7435 *(0,0585)*	26,6522 *(0,0091)*	199.120	82.242	89.759	11.815	12.307
I	80%	dec	1	1	9,8410 *(0,0121)*	50,1138 *(0,0171)*	223.490	106.570	91.252	13.240	12.434
II	80%	dec	1	0,1	9,4317 *(0,0116)*	49,6723 *(0,0170)*	223.900	106.710	91.465	13.264	12.462
III	γ	dec	1	1	47,0877 *(0,0577)*	29,0785 *(0,0099)*	200.210	85.374	90.445	11.950	12.437
IV	γ	dec	1	0,1	48,6998 *(0,0597)*	28,7983 *(0,0098)*	200.430	85.490	90.556	11.957	12.427

Table 15: *Summary of cost function values in different scenarios with different weights.*

not consider the economical term into the cost function. However, the simulator could compute the value for the economical term, and computes the cost. In Table 15 the values, computed using the SIMULINK® model, regarding the term in the cost function are reported. Eight simulation have been done. In particular, four using the centralized and four using the decentralized. For both controllers the weights chosen for the security and the stability term are in the first simulation one and one, while in the second one and zero point one. In all the simulations the initial condition is always the same: the 50% of the maximal volume, for this reason it is not reported in the table. The penalty volumes used are two: the 80% and the value reported in the first column in Table 12, which represents the picks of the minimal volume necessary to satisfy the demands, this choice in Table 14 is represented with the letter γ. In Table 15, the total value obtained in the stability and security term is also reported, in order to compare the data coming from the centralized and decentralized in a more simply and clear way, the normalized value is into the round bracket. The use of a decentralized control, which considers the network formed by three different subsystems, causes a small worsening of every objective term.
It is possible to compare the results reported in the Tables 14 and 15 only regarding the stability and the economical terms. Indeed, the penalty term is computed differently in PLIO and in the control implemented in MATLAB®. The results, in terms of economical cost in euros, in the Table 15 are very big since does not considers the economical term in the optimization. These results are comparable to those obtained in the last scenarios in the Table 14, where the economical term does not have the main priority in the optimization. These values could be considered as a upper bound for the price in this network with these particular choice of the parameters.

7 Conclusions and Future Work

7.1 Conclusion

In this report two different combined works have been presented. The first aims to the implementation of a simulation water network environment which can be used to evaluate the best parametrization of a MPC controller of water transport network.

In particular, this environment is developed in SIMULINK®/MATLAB® and is able to reproduce the behaviour of a water network, and it could be interfaced with whatever controller. It represents an useful tool to evaluate the performance and the operation of any controller. It allows to show the best choice of the MPC parameters in terms of cost function. In addition, to check if the controller satisfies all the constraints presented in the model, as the satisfaction of the demands or the mass balances.
In particular, the case of the aggregated Barcelona water network has been studied. The SIMULINK® model for the Barcelona case has been developed and it has been interfaced with PLIO tool [5] and with the centralized and decentralized MPC controllers developed in [7].

7.1.1 Analysis using PLIO Tool

The PLIO tool allows to implement a centralized controller for water networks with the same ideas of the MPC with some variations which allow to render the cost function more close to the real request. Using this type of controller, which uses the commercial solver GAMS, we have developed an MPC controller and a studied the effect of the initial conditions, penalty volumes and weights in the cost function. This analysis has been done thanks to the results obtained from the simulator which allows to compute the cost function term in a straightforward way. For every of the three elements, we have obtained some conclusions, which allow us and the user in general, to understand how to tuning the MPC controller:

Penalty volume : the minimal penalty volume which manages to satisfy the demands, the main objective of the controller, is determined. The controller is optimal when then minimal penalty volume is used. To compute this penalty volume, the minimal volume necessary to satisfy the demands at every time in a scenario quite general has been computed. After, in order to obtain a constant penalty level, with a degree to security, we have computed the picks for every tanks and added to this a 20%.

Initial condition : first of all we have noticed that the initial condition influences a lot the cost of the water network operation, in particular in the first day. Moreover, our study has been concentrated also to search a particular initial condition which allows to obtain the same evolution every day, but this issue need some further investigations.

Weights : the weights study includes the importance order of both the elements and the cost function objectives.
Concerning the elements we have decided to weight more the valve which follows a source. Indeed the actuators used during the water treatments need a major stability in the setpoint respect with the others.
The objectives of the cost function considered in this study are three: the stability of the actuators, the economical price and the security level in the tanks. We have found out, through the simulations reported in the Table 14, that the economical term have to be the main important objective in the cost function in order to obtain a minimal economical cost, and so to save money. The other two weights could be decided freely according by the particular situation considered.

7.1.2 Analysis using MPC Controller

The other part of the study is concentrated on the interfacing of the simulator with the controller developed in the [7]. Also in this case, several different simulations has been realized which allow to highlight the different performances obtained with a centralized and a decentralized controller. These simulations are reported in Table 15. This type of controller does not include the economical term, and it is the cause of the big cost obtained, comparing to that obtained through PLIO when the economical term is the term with the minor importance. Moreover, the security term it is computed in a different way.

All these analysis proved the importance of a predictive controller in this water network. The predictive controller could be of various types, with a correct parametrization, it manages always to obtain a better solution with respect to that obtained with a manual regulation.

7.2 Future Works

This study has been developed for the aggregated model of the Barcelona water network. The same study should be made for the complete water network reported in the Figure 5, where, although the number of the variables increase a lot as well as the complexity, the strategy used in the development of the simulator and in the parameters modelling of the network remains the same.

To make the model, and the simulation, more realistic and useful for the daily use it could be necessary using several parameters with a degree of major accuracy, or in general with several characteristics that allow to reproduce the reality. In particular the study would be concentrated in the realization of a controller which should have the security volume in the tank not as a constant level but it could be an evolution which respects the profile of the demand in time. It could be something similar to that showed in the Figure 27 in the broken red line, but this figure represents the demands and the volume with a particular scenario. Instead, it would be important to use an universal profile, valid for every simulation. In addition, it would be interesting to define a different profile for every day of the week, which reflects the differences between the day type (working or non working days). In order to obtain a standard and predictable behaviour of the network the study concerning the search of a "repetitive" initial condition should be continued.

Another thing that is missing in the model is the consideration of the pressure effect in the water network, in order to study a real hydraulic model with its constraints. For the moment, the pressure constraints have been considered regulating the flow in the actuators, but it would be necessary a more complex model.

The simulator developed would be applied to a whatever water network, but to simplify the building it would be required an automatic generation of the blocks through a script file. The connections would be created, directly, since the state space description (the matrix A, B, C and D) generated and used in the MPC controller. At this point, it should be developed a downright TOOLBOX for the hydraulic model. This toolbox would be included a different block for every different elements that could be present in a water network. Generating the SIMULINK® network in an automatic method the building of a simulator of a water network become simply and fast. Moreover, this simulator should be closed in a feedback loop with the controller, in order to get a evaluation of the performance on-line instant by instant, which allows to influence the controller decision at the next step.

References

[1] *Manual de Usario PLIO*, v12 edition, 06 2005.

[2] M.A. Brdys and B. Ulanicki. *Operational control of water systems: structures, algorithms, and applications*. Prentice Hall, 1994.

[3] E.F. Camacho and C. Bordons. *Model Predictive Control*. 1999.

[4] EF Camacho and C. Bordons. *Model Predictive Control*. Springer, 2004.

[5] G. Cembrano, J. Quevedo, V. Puig, R. Pérez, and J. Figueras. Predictive Optimal control of water supply and distribution in Santiago-Chile.

[6] G. Cembrano, G. Wells, J. Quevedo, R. Prez, and R. Argelaguet. Optimal control of a water distribution network in a supervisory control system. *Control Engineering Practice*, 8(10):1177–1188, 2000.

[7] V. Fambrini and C. Ocampo-Martinez. Modelling and Decentralized Model Predictive Control of Drinking Water Networks. Technical report, Institut de Robòtica i Informàtica Industrial, CSIC-UPC, 2009.

[8] RW Jeppson. *Analysis of flow in pipe networks; R.* Ann Arbor Science Publishers, Ann Arbor, New York, 6^{th} edition, 1983.

[9] R.W. Johnson. *The handbook of fluid dynamics*. Springer, 1998.

[10] J.M. Maciejowski. *Predictive Control: With Constraints*. Prentice Hall, 2002.

[11] E. Mosca and P. Optimal. Adaptive Control. *PrenticeHall, Englewood Clis, NY*, 1995.

[12] V. Nitivattananon, E.C. Sadowski, and R.G. Quimpo. Optimization of water supply system operation. *Journal of Water Resources Planning and Management*, 122(5):374–384, 1996.

[13] Carlos A. Ocampo-Martinez. *Model Predictive Control of Complex Systems including Fault Tolerance Capabilities: Application to Sewer Networks*. PhD thesis, Automatic Control Department (ESAII) Technical University of Catalonia, April 2007.

[14] Sociedad general de Agua de Barcelona, S.A. (AGBAR), Passeig de SAnt Joan, 39-43. *Sistema Automtico de Explotacin (SAE)*, March 1999.

[15] M.Y. Tu, T.C.T. Frank, W.G.Y. William, et al. Optimization of water distribution and water quality by hybrid genetic algorithm. *Journal of water resources planning and management*, 131:431, 2005.

[16] K.S. Westphal, R.M. Vogel, P. Kirshen, and S.C. Chapra. Decision support system for adaptive water supply management. *Journal of Water Resources Planning and Management*, 129(3):165–177, 2003.

IRI reports